Fachhochschule Köln
Cologne University of Applied Sciences

Eine vertrauenswürdige Architektur zum Teilen nutzergenerierter Inhalte im Web

Master Thesis

vorgelegt an der Fachhochschule Köln
Campus Gummersbach
im Studiengang Medieninformatik

ausgearbeitet von:

Tim Schneider B.Sc. (11042662)
tim.schneider@smail.fh-koeln.de

Gummersbach, im Januar 2011

Eine Abschlussarbeit zur Erreichung des akademischen Grades
„Master of Science“

Erstprüfer: Prof. Dr. Kristian Fischer
Zweitprüfer: Prof. Christian Noss

Zusammenfassung

Die Dynamik des *Social Webs* motiviert zum Teilen nutzergenerierter Inhalte. Diese entstehen in zahlreichen Social Networks meist unter Missachtung der Schutzziele der IT-Sicherheit: Vertraulichkeit, Verfügbarkeit und Integrität von Nutzerdaten. Betreiber von Web-Anwendungen können Inhalte ihrer Nutzer einsehen, fälschen, löschen oder zu unbekannten Zwecken auswerten und verfügen über Wissen über Kommunikationspartner und -verhalten — ohne, dass sich Benutzer wirksam davor absichern könnten. Von dem im Grundgesetz verankerten *Recht auf Privatsphäre* ausgehend soll im Rahmen dieser Ausarbeitung eine neuartige Architektur zum Teilen nutzergenerierter Inhalte im Web entwickelt werden, die Benutzeranforderungen an die Erfüllung der Schutzziele der IT-Sicherheit vollständig gewährleistet und darüber hinaus durch eine bewusste Kommunikation dieser Qualität als *vertrauenswürdig* aufgefasst werden kann. In einem *Goal-directed Design*-Prozess wird eine Architekturskizze entwickelt, welche die im Prozess erarbeiteten Benutzeranforderungen durch die Bereitstellung zweier Web-Services erfüllt: Der *Signed Content Storage* adressiert als zuverlässiger und durch den Urheber autorisierter Web-Speicherort signierter, nutzergenerierter Inhalte die Schutzziele Verfügbarkeit und Integrität. In Kombination mit dem *Identity Provider*, der gesicherte Informationen von Urheber und Teilhabern zur Verfügung stellt, ist ein vertrauliches Teilen von Inhalten im Web möglich. Vertrauenswürdigkeit gewinnt diese Architektur durch konsequente Transparenz, Selbstbeschreibungsfähigkeit, externe Bewertbarkeit und der Dokumentationsfähigkeit von Nutzungserfahrungen.

Begriffsbestimmungen

Eine vertrauenswürdige Architektur zum Teilen nutzergenerierter Inhalte im Web. Grundlegend enthält der Titel dieser Ausarbeitung den Begriff des *Webs*. Für das weitere Verständnis und zur Einordnung der Ausarbeitung in einen thematischen Kontext sollen dieser Begriff und drei darauf aufbauende Software-Konzepte einleitend schärfer definiert werden.

Web Der Begriff *Web* wird im Folgenden synonym zum *World Wide Web* genutzt und bezeichnet den Technik- und Kulturraum des über das Internet aufrufbaren Hypertext-Systems als universale Plattform für Web-Anwendungen und -Services.

Web-Anwendung Eine *Web-Anwendung* ist ein Computer-Programm, welches auf einem Web-Server ausgeführt wird. Über das Web (den Browser und die grafische Benutzungsschnittstelle der Anwendung) können Benutzer mit der Anwendung interagieren. Im Gegensatz zu einer statischen Webseite besteht der Zweck einer Anwendung nicht aus der reinen Informationsvermittlung durch die Darbietung von Hypertexten, sondern darin, Benutzer durch angebotene Funktionalitäten bei der Erreichung persönlicher Ziele zu unterstützen. Moderne Web-Anwendungen stellen dabei häufig spezialisierte Funktionalitäten für einen scharf umrissenen Anwendungsfall zur Verfügung und fokussieren eine bestmögliche Unterstützung einzelner (Geschäfts-)prozesse ihrer Benutzer.

Web-Service Der Begriff *Web-Service* wird in der öffentlichen Diskussion häufig als Analogon zur vorgenannten *Web-Anwendung* genutzt. Es existieren auch technische Ähnlichkeiten — bei beiden Konstrukten handelt es sich um Software, die auf einem Web-Server ausgeführt wird und über das Web zugänglich ist —, jedoch soll ein grundlegender Unterschied die Begriffe im Kontext dieser Ausarbeitung stärker voneinander trennen: Während eine Web-Anwendung die Interaktion mit und die Prozessunterstützung von Personen in den Vordergrund stellt, definiert das World Wide Web Consortium (W3C) die „Bereitstellung eines Webservices als Unterstützung zur Zusammenarbeit zwischen verschiedenen Anwendungsprogrammen“ und stellt dabei die „direkte Interaktion mit anderen Software-Agenten“ heraus[1] —

[1] vgl. [W3C04]

es handelt sich hierbei also um eine Maschine-zu-Maschine-Kommunikation über Programmierschnittstellen. Vereinfacht gesprochen kann man einen Web-Service als Web-Anwendung verstehen, ohne eine für eine menschliche Interaktion entworfene Präsentationsschicht.

Architektur einer Web-Anwendung Als die Architektur einer Web-Anwendung wird der der Anwendung zu Grunde liegende Aufbau und die Organisation der an der Anwendung beteiligten Systeme bezeichnet. Dies schließt insbesondere die Organisation der Anwendungslogik und der Datenhaltung ein. Mit der Präsentationsschicht als drittes Konstrukt wird diese Perspektive einer Dreiteilung der Systemkomponenten auch als *3-Tier-Architektur* bezeichnet. Diese Architekturform ist im Web weit verbreitet und stellt den Ausgangspunkt der Problembetrachtung innerhalb dieser Ausarbeitung dar.

Gliederung

1 Einführung

«Web 1.0 was all about connecting people. It was an interactive space, and I think Web 2.0 is of course a piece of jargon, nobody even knows what it means. If Web 2.0 for you is blogs and wikis, then that is people to people. But that was what the Web was supposed to be all along.»

— Tim Berners-Lee, im Juli 2006

Auch wenn es bereits 1990 Tim Berners-Lees Motivation und Vision war, über das Web Menschen miteinander zu verbinden, weltweite Kommunikation zu ermöglichen und dass das Web von Anfang an in gleichen Maße zum Publizieren wie zum Konsumieren von Inhalten genutzt werden sollte, erst 15 Jahre später, als Tim O'Reilly im September 2005 seine Beobachtungen zu den damals aktuellen Entwicklungen des World Wide Webs verschriftlichte und mit seinem Artikel den viel diskutierten Begriff *Web 2.0* prägte, begann man allgemein zu begreifen, dass Berners-Lees Idee und Vision des partizipatorischen World Wide Webs sich mehr und mehr zu verwirklichen schien. O'Reilly beschrieb das *Web 2.0* als eine „Veränderung in der Geschäftswelt" und als eine „neue Bewegung in der Computerindustrie hin zum Internet als Plattform". Wesentliche Veränderungen waren aus seiner Sicht:

- Das Web als Plattform ersetzt Software auf dem lokalen Rechner der Web-Nutzer und lenkt Aufmerksamkeit ins Web
- datengetriebene Anwendungen fokussieren die Erstellung und gemeinsame Nutzung so genannter *nutzergenerierter Inhalte* im Web
- eine „Architektur des Mitwirkens" ermöglicht es und motiviert Benutzer(n) im Web aktiv zu werden; hierbei ist besonders die gesteigerte Interaktivität von Web-Anwendungen herauszustellen, die sich durch neuartige Nutzungen bereits vorhandener Web-Techniken ergab[1]
- Mash-Up-Anwendungen nutzen Programmierschnittstellen, um Daten-Pools untereinander zu verbinden und so neue Nutzungsmöglichkeiten separierter Daten zu schaffen

[1] das AJAX-Konzept (Abkrz. für „Asynchronous JavaScript and XML") beispielsweise, welches asynchrone HTTP-Anfragen im Browser beschreibt, geht auf eine Remote-Scripting-Komponente von Microsoft zurück, die bereits 7 Jahre in der Internet Explorer-Reihe verfügbar war bis Jesse James Garrett 2005 ihr Potenzial erkannte und einen Aufsatz dazu veröffentlichte, der schließlich größere Aufmerksamkeit auf sich zog

- einfache Geschäftsmodelle werden mit fokussiert entwickelter Software kombiniert, deren Fähigkeiten nicht über den einzelnen Verwendungszwecks hinausgehen und die im Dialog mit ihren Benutzern ständig weiterentwickelt wird

So bezeichnet der Begriff *Web 2.0* nach Tim O'Reillys Verständnis nicht eine spezifische technische Entwicklung, sondern vielmehr eine eine veränderte Nutzung und Wahrnehmung des Internets.

Heute, weitere 5 Jahre später, hat der viel diskutierte Begriff an Aktualität nicht verloren, auch wenn seine Verwendung zugunsten von *Social Media* abnimmt, was als Oberbegriff für alle Arten sozialer Netzgemeinschaften verstanden wird, die im Wesentlichen Plattformen des gegenseitigen Austauschs sind.

1.1 Situation

Menschen nutzen das Web. Sie erstellen Inhalte mit dem und für das Web. Manche von Ihnen würden behaupten, sie *leben* gar im Web. Sie produzieren Podcasts[2], andere schreiben Texte[3]. Sie laden Videos ins Web[4] und erstellen aus Urlaubsfotos musikuntermalte Slideshows für ihre Bekannten und Unbekannten. Sie sammeln interessante Links zu seltenen Themen und kommentieren und bewerten diese öffentlich. Sie zeigen anderen wen und woher sie sie kennen und was und wo sie studiert haben[5]. Sie veröffentlichen Lebensläufe, teilen Kochrezepte[6] und Wissen[7]. Sie veranstalten Videokonferenzen mit den Mitarbeitern der Unternehmenszweigstelle in Stockholm und überarbeiten dabei gemeinsam die Quartalszahlen. Sie schreiben den Brief an den Vermieter online[8] — zusammen mit dem Mitbewohner, der gerade in London ist[9]. Sie nehmen an E-Petitionen des Deutschen Bundestages teil[10], twittern morgens ihrem Wahlkreiskandidaten Zuspruch und abends in welchem Club sie gerade feiern und dass sie nun verlobt sind[11]. Die Bilder des Rings schauen sich die Freunde bei Facebook an. *„I like"* — das Web ist Alltag.

[2] und manche erreichen damit bis zu 30.000 Hörer pro Episode [Pri10]
[3] und erhalten dafür einen Grimme-Preis [Hae06]
[4] und bereits 2008 schaute lt. [WO08] jeder dritte Amerikaner täglich eines davon an
[5] im nach eigenen Angaben „größten Social Network in Europa" StudiVZ tun dies aktuell über 16 Millionen Studenten; die meisten von ihnen kommen aus dem deutschsprachigen Raum [Cru10]
[6] u.a. im freien, deutschen Rezepte-Wiki [RWC10]
[7] lt. Wikipedia-Gründer Jimmy Wales nutzen aktuell 400 Millionen Menschen jeden Monat die über 10 Millionen Einträge der freien Online-Enzyklopädie [Wal10]
[8] http://piratepad.net/sxTNfxeCXr
[9] exakte Position bekannt: Russel Square Gardens; 51.52123, -0.12669
[10] Petition: Internet - Keine Indizierung und Sperrung von Internetseiten vom 22.04.2009, [Hei10]
[11] „Romantik 2.0 - Ein Heiratsantrag auf Twitter" [Her09]

Es bietet scheinbar unendlich viele Möglichkeiten es aktiv mitzugestalten und Informationen über sich, über andere und über alles andere zu veröffentlichen, zu verändern, zu bewerten, zu kommentieren — eben **zu teilen**. Mit der starken Verbreitung mobiler Internet-fähiger Geräte (Smartphones, Netbooks, ...) haben Menschen ihren leistungsfähigen Zugang zum World Wide Web zudem auch unterwegs dabei und das Teilen von Informationen, Eindrücken, Gedanken, Fotos oder Videos kann jederzeit von fast jedem Ort aus spontan, direkt und bidirektional stattfinden. Den Zustand „offline“ erlebt die Generation, die mit dem Web aufgewachsen ist und in der Literatur auch als *Digital Native* bezeichnet wird, daher kaum noch. Für viele Menschen ist es selbstverständliche eine ständige Verbindung zum persönlichen sozialen (Online-)Netzwerk zu haben und über dessen Aktivitäten auf dem Laufenden zu bleiben.

Eine im September 2010 von MTV Networks vorgestellte, international durchgeführte Studie zur Mediennutzung durch Jugendliche beziffert den Anteil der *Digital Natives*[12], die täglich soziale Online-Netzwerke nutzen, auf 58 Prozent [Net10]. Im Schnitt 119 Minuten am Tag schreiben sie sich Nachrichten auf Pinnwände, kommentieren die Beiträge anderer, schauen Videos und Fotos ihrer Freunde an. Einen Eindruck davon wie viele Daten dabei entstehen können, vermitteln die öffentlichen Statistiken von Facebook: Allein dort teilen über 500 Millionen Mitglieder pro Monat rd. 30 Milliarden *neue* Inhaltsobjekte — Tendenz: steigend [Fac10].

Doch hat die Dynamik der *Social Media*-Wolke nicht nur Implikationen auf die private Nutzung des Webs. Durch die steigende Popularität von Social Software auch im geschäftlichen Kontext und die Entstehung themenvertikaler *Business Social Networks* entwickelten sich mit dem *Web 2.0* auch neue Arten der Arbeit und Zusammenarbeit. Jene Web-Anwendungen adressieren besonders die Bedürfnisse und Nutzungsszenarien von Unternehmen: Ob web-basierter Dokumentenaustausch[13], Office-Anwendungen[14] , Mindmapping-Werkzeuge[15], Kontakt-Netzwerke[16] oder Zeiterfassungssysteme[17]; als Abo-Modell *„Software as a Service“* (SaaS) angeboten profitieren vor allem kleinere und mittelständische Unternehmen von deren überall[18] verfügbarer Funktionalität. Anwendungen, die extern betrieben, gewartet und weiterentwickelt werden, sind zudem ohne großes Initialinvestment (wie es bei einer Eigenentwicklung der Fall wäre) direkt nutzbar und dennoch durch Benutzer-Feedback

[12] Personen, die im Jahr 2010 zwischen 14 und 29 Jahren alt sind
[13] z.B. *Dropbox*, https://www.dropbox.com/
[14] z.B. *Google Docs*, http://docs.google.com/
[15] z.B. *MindMeister*, http://www.mindmeister.com/de
[16] z.B. *XING*, https://www.xing.com/
[17] z.B. *mite.*, http://mite.yo.lk/
[18] im Sinne von: die Software ist verfügbar, sobald ein Gerät auf das World Wide Web zugreifen kann und einen kompatiblen Browser besitzt

und die stetige Weiterentwicklung durch die Betreiber mittelfristig an eigene Anforderungen anpassbar. Während so in der Anfangszeit des World Wide Webs die geschäftliche Internet-Kommunikation, die über die Nutzung von E-Mail hinaus ging, meist eher im unternehmenseigenen Intranet stattfand, welches auch aus Sicherheitsbedenken vom World Wide Web getrennt war, sind es nun die Eigenschaften des Webs *Universalität*, *Verfügbarkeit* und *Mobilität* und das SaaS-Geschäftsmodell, die es zu einer attraktiven Architektur für geschäftlich genutzte Software machen, neue Formen der computergestützten Zusammenarbeit ermöglichen und Geschäftsdaten und deren Verarbeitung ins Web verlagern.

1.2 Problem

Aus der Bereitschaft und dem Wunsch der Menschen eigene Inhalte im Web miteinander zu teilen und dem großen Angebot dies in verschiedensten thematischen Kontexten und multimedialen Ausprägungen tatsächlich täglich tun zu können, entwickeln sich neue, spannende Kommunikationsformen, Lebens- und Geschäftsmodelle.

Diese Dynamik verlangt eine genauere Betrachtung der technischen Grundlagen des Teilens und jener für viele Web-Anwendungen typischen 3-Tier-Architektur, die aus zentralen Datenbankservern (Datenhaltungsschicht) und zentralen Anwendungsservern (Logikschicht)[19] besteht (Abbildung 1.1). Sie birgt zahlreiche Probleme beim Umgang mit nutzergenerierten Inhalten im Web: Denn egal ob Texte, Bilder, Musik, Videos, Geodaten, Kontaktdaten, Programmcode oder Links — so verschieden die Inhalte auch sein mögen, haben sie eines immer gemeinsam: Sie entstehen und existieren **im Web**. Diese Selbstverständlichkeit kann für deren Nutzung auch nachteilig sein.

Viele Web-Anwendungen basieren auf der in Abbildung 1.1 dargestellten architektonischen Dreiteilung, auch wenn sie ggfs. mehrere Systeme pro Schicht parallel einsetzen (z.B. verteilte Datenbanksysteme oder parallel arbeitende Anwendungsserver zur Lastverteilung). Die in der Abbildung orange umrandeten Elemente werden durch den Betreiber der Anwendung kontrolliert, sie befinden sich unmittelbar und ständig *im Web*. Die Client-Rechner kommunizieren über deren Schnittstelle, z.B. mittels eines Web-Browsers.

[19] vervollständigt wird eine 3-Tier-Architektur schließlich durch eine Präsentationsschicht, die sich bei Web-Anwendungen zum Teil auf Server-Seite (HTML/CSS), zum Teil auch auf Client-Seite (JavaScript) befinden kann; die Präsentationsschicht ist für diese Problembetrachtung jedoch zunächst nicht relevant

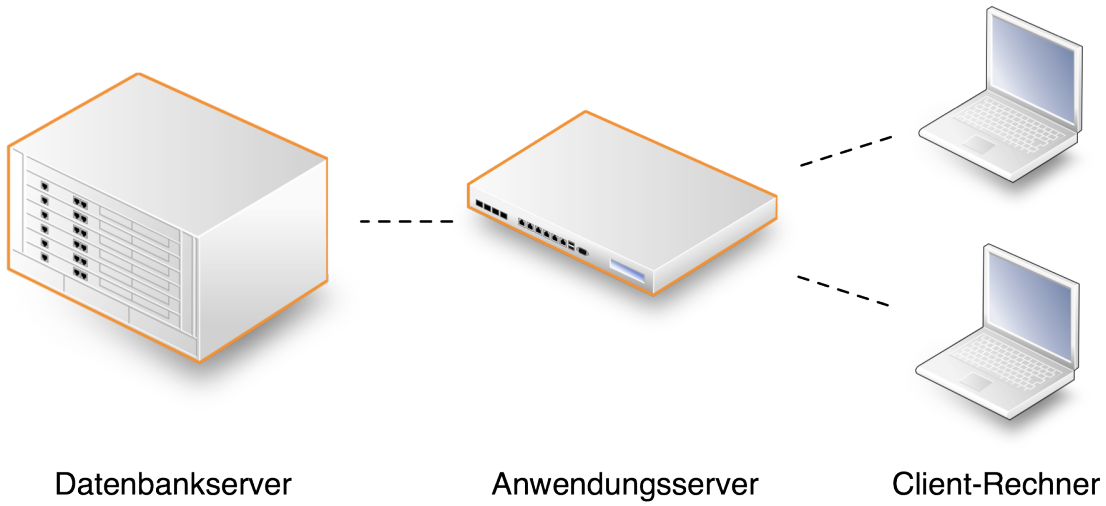

Abbildung 1.1: Eine 3-Tier-Architektur besteht aus Datenhaltungsschicht (Datenbankserver), Logikschicht (Anwendungsserver) und Präsentationsschicht (Anwendungsserver/Client-Rechner).

Ausgehend von dieser Situation soll die weitere Problem-Betrachtung anhand einer typischen Kommunikation verdeutlicht werden, wie sie in sozialen Online-Netzwerken (*Social Networks*) stattfindet. Dort besitzen die Benutzer häufig eigene innerhalb des Netzwerkes öffentliche Profile, auf denen andere Benutzer Nachrichten hinterlassen können. Diese Funktion wird auch als die „Pinnwand“ eines Profils bezeichnet. Nachrichten können dabei häufig multimedial sein und dienen als universeller Container für gemeinsam geteilte Inhalte.

Um eine Pinnwand-Nachricht zu hinterlassen, stellen Web-Anwendungen über die Präsentationsschicht üblicherweise Formulare zur Verfügung mit Texteingabe- und ggfs. auch Hochlademöglichkeiten für lokale Dateien. Ein Benutzer verfasst nun direkt im Browser einen Text und sendet das Formular ab. Der Browser schickt daraufhin die Formulardaten als Bestandteil einer HTTP-Anfrage an die öffentliche Schnittstelle der Web-Anwendung. Die Anwendungslogik verarbeitet die Anfrage und die darin enthaltenen Formulardaten, erstellt daraus ein neues Inhaltsobjekt (zunächst nur im Arbeitsspeicher des Servers) und delegiert dessen persistente Speicherung an ein angeschlossenes Datenbanksystem. Dieses veranlasst die physische Persistierung auf einem Datenträger und kann das gespeicherte Objekt fortan auf Anfrage der Anwendungslogik von diesem lesen und zur Verarbeitung herausgeben. Dies ist zum Beispiel dann der Fall, wenn der Besitzer der digitalen Pinnwand die Nachrichten seiner Pinnwand lesen möchte und diesen Wunsch über das Klicken eines Links im Frontend der Anwendung (Präsentationsschicht) und dadurch über eine entsprechende HTTP-Anfrage an die Anwendungslogik übermittelt (Abbildung

1.2).

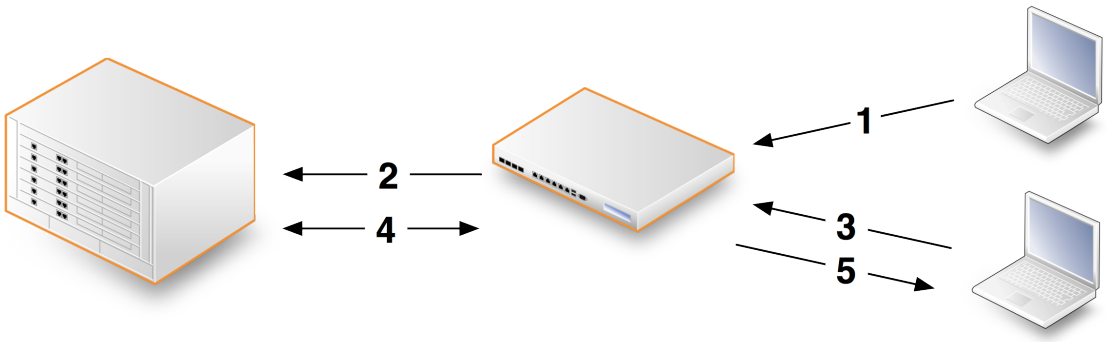

Abbildung 1.2: Der Weg einer Pinnwand-Nachricht. Alice schickt diese über ihren Browser an die Anwendungslogik (1), diese leitet sie zur Speicherung an das Datenbanksystem weiter (2). Bob fragt die Nachricht über seinen Browser an (3), die Anwendungslogik fragt diese beim Datenbanksystem an (4), erhält sie (4) und übermittelt sie schließlich an den Browser von Bob (5).

Der Kommunikationsweg vom lokalen Rechner bis zur Schnittstelle der Web-Anwendung, über den Benutzer-Anfragen und damit auch Benutzer-Inhalte übertragen werden, ist das Web (Pfeile 1, 3 und 5 in Abbildung 1.2) und er ist damit für einen Benutzer technisch nachvollziehbar und in Bezug auf die Sicherheit der übertragenen Inhalte einschätzbar. Jedoch wo genau im Anschluss daran, auf welchen physischen Systemen der Web-Anwendung dabei tatsächlich temporär und schließlich auch persistent die Inhalte des Benutzers übertragen und gespeichert werden, und ob außer dem Urheber (Alice) und weiteren durch ihn autorisierten Personen (Bob) unbekannte Dritte (ein Administrator der Anwendung?) logischen oder physischen Zugriff auf die Inhalte besitzen, dies bleibt in der Regel unklar: Die der Web-Anwendung zugrundeliegende Architektur besitzt aus der Perspektive des Benutzers *Blackbox*-Charakter, da alle Verarbeitungsschritte, die nach einer Anfrage an einen Anwendungsserver geschehen, nicht nachvollziehbar sind. Bei stark virtualisierten Systemen und in SOA-Kontexten[20], in denen die einzelnen Schichten der Web-Server-Seite gekapselt und über Dienste miteinander verbunden sind[21], und somit auch untereinander als Blackbox wahrgenommen werden, ist das physische Geflecht aller beteiligten Systeme, deren Administratoren und Zugriffsmöglichkeiten nur für wenige Personen überhaupt ergründbar.

[20] dienstorientierte Architektur, engl. *Service-oriented architecture*. Ein „Paradigma für die Strukturierung und Nutzung verteilter Funktionalität“ (OASIS, 2006).
[21] z.B. bei Nutzung des Amazon Web Services S3 als Datenhaltungsschicht

1.2.1 Vertraulichkeit von Daten

Dass Daten in Informationssystemen lediglich von autorisierten Personen gelesen und verändert werden können, bezeichnet man im Vokabular der IT-Sicherheit als die **Vertraulichkeit von Daten** [SB92]. Diese wird im Rahmen der **Informationssicherheit** als eines der Schutzziele personenbezogener Daten angeführt. Im Sinne der IT-Sicherheit, sind Informationssysteme so zu gestalten, dass die Gewährleistung der Schutzziele in ausreichendem Maße sichergestellt wird.

Web-Anwendungen, bei denen die Verarbeitung und Speicherung von Benutzerdaten aus Benutzersicht in einer Blackbox geschehen und dadurch nicht nur die tatsächlichen Verarbeitungs- und Speicherungsorte unbekannt sind, sondern auch die Gruppe an Personen, die zu Administrationszwecken (oder auch einfach „weil sie es können“), Zugriff auf die Systeme und Benutzerdaten besitzen, können allein aufgrund dieser Architektur das Schutzziel der Vertraulichkeit der Daten nicht vollständig erfüllen.

Aufgrund der zentralen, durch einen einzelnen Betreiber kontrollierten Architektur ist zudem eine Manipulation nutzergenerierter Inhalte durch den Betreiber denkbar. Dies wird offensichtlich, betrachtet man private Nachrichtensysteme in Web-Anwendungen. Schreibt Alice private Nachrichten an Bob, werden diese meist innerhalb der Anwendung selbst verfasst, verschickt, zugestellt, gelesen und abgelegt. Über den gesamten Lebenszeitraum einer Nachricht wird diese somit *nicht* von Alice oder Bob, sondern allein durch den Betreiber der Anwendung kontrolliert. Sowohl Absender als auch Empfänger üben über die von der Anwendung zur Verfügung gestellten Funktionen lediglich eine indirekte Kontrolle aus. Ungeachtet ihres Einverständnisses können Nachrichteninhalte nach dem Absenden gekürzt, gelöscht oder ergänzt werden. Nachrichten könnten strategisch verzögert, an nicht durch den Absender autorisierte Empfänger oder auch gar nicht zugestellt werden.

1.2.2 Integrität von Daten

Dass Daten in Informationssystemen nicht unbemerkt verändert werden können bzw. alle Änderungen nachvollziehbar sein müssen, wird als die **Integrität von Daten** bezeichnet. Diese ist insbesondere dann gefährdet, wenn von verschiedenen Seiten (Systeme oder Personen) auf Informationen zugegriffen und sie verändert werden können. Neben dem Schutzziel der Vertraulichkeit, kann auch die Erreichung der Integrität nutzergenerierter Inhalte durch eine zentrale, vom Benutzer nicht

kontrollierbare Architektur nicht gewährleistet werden[22].

Schließlich führt die dauerhafte Speicherung eigener Inhalte auf fremden Systemen zu einer eingeschränkten Verfügbarkeit der Inhalte: Ist eine Web-Anwendung überlastet oder werden Wartungsarbeiten durchgeführt sind eigene Inhalte temporär nicht verfügbar. Wird gar der Betrieb einer Web-Anwendung eingestellt, innerhalb derer Benutzer zuvor Inhalte erstellt und gespeichert haben, oder erleidet diese zerstörerische Angriffe, kann ein kompletter Datenverlust die Folge sein.

1.2.3 Verfügbarkeit von Daten

Unter der **Verfügbarkeit von Daten** versteht man die jederzeitigen Zugriffs- und Nutzungsmöglichkeiten von Daten durch dessen Urheber. Als drittes Schutzziel der IT-Sicherheit schreibt es vor, dass der Zugriff auf eigene Daten innerhalb eines vereinbarten Zeitrahmens zu gewährleisten ist. Es liegt jedoch in der Natur des Webs, dass eine uneingeschränkte Verfügbarkeit entfernt abgelegter Daten nicht vollständig gewährleistet werden kann. Selbst umfangreiche, mehrfach redundante Enterprise-Storage-Systeme garantieren keine 100-prozentige Verfügbarkeit[23].

1.2.4 Ein Architekturproblem

Der Erreichung aller Schutzziele

- Vertraulichkeit,
- Integrität und
- Verfügbarkeit

von nutzergenerierten Inhalten steht eine zentralistisch organisierte Architektur von Web-Anwendungen unmittelbar im Wege. Es handelt sich um ein systemimmanentes Problem: Solange die Weitergabe und Speicherung eigener Inhalte nicht über ihren gesamten Lebenszeitraum durch ihren Urheber allein kontrolliert werden können, sondern fremde Dritte unbekannte Arten von Zugriff auf die an der Weitergabe und Speicherung beteiligten Systeme besitzen, ist eine zweifelsfreie Gewährleistung

[22] zumindest nicht ohne wirksamer Ende-zu-Ende-Verschlüsselung aller Benutzerdatenströme. Auf dieses Problem wird in Abschnitt 1.2.4 näher eingegangen.

[23] vgl. dazu u.a. „Unsere Service Level Agreements garantierten Ihnen u.a. eine Verfügbarkeit von über 99,9%.“, Host Europe GmbH [Eur10]

aller Schutzziele nicht möglich. Sämtliche Web-Anwendungen, die auf einer derartigen Architektur basieren, können in der Folge nicht zum Zwecke **vertraulicher Kommunikation** genutzt werden.

Denn selbst, wenn innerhalb eines fremdkontrollierten Systems eine wirksame Ende-zu-Ende-Verschlüsselung aller Nutzerkommunikation stattfindet[24] und somit ein inhaltlicher Zugriff auf oder die Manipulation der Nutzerinhalte durch Dritte wirksam verhindert werden kann, so bleibt beim Anwendungsbetreiber dennoch das **Wissen über die Kommunikation** selbst, über die Kommunikationspartner, deren -frequenz und -dauer. Auch diese (Meta-)Informationen sind als nutzergenerierte Inhalte anzusehen, können vertraulich sein und sind ebenso schützenswert wie die eigentlichen Kommunikationsinhalte. Ferner bestände auch bei einer Ende-zu-Ende-Verschlüsselung weiterhin die Möglichkeit die Datenübermittlung spezifischer Benutzer zu manipulieren. Dass Datenströme unterschiedlicher Nutzer im Internet durch ihre Vermittler unterschiedlich priorisiert werden, bezeichnet man als einen Angriff auf die *Netzneutralität*[25], die auch in Web-Anwendungen gelten sollte.

1.2.5 Bedeutung der Schutzziele

Die Schutzziele der IT-Sicherheit werden aktuell in vielen Web-Anwendungen nicht vollständig gewährleistet. Wenige Anwendungen bieten ihren Nutzern die Verschlüsselung der Übertragung eigener Inhalte an[26], um damit unautorisiertem inhaltlichen Zugriff vorzubeugen. Noch weniger Anwendungen legen die Inhalte ihrer Nutzer tatsächlich verschlüsselt ab. Betreiber von Web-Anwendungen argumentieren oft damit, dass eine vollständige Verschlüsselung einen sehr hohen technischen Aufwand bedeute, der nicht wirtschaftlich zu rechtfertigen sei[27]. Es ist allerdings anzunehmen, dass auch das Wissen um die Inhalte der eigenen Benutzer wirtschaftlich verwertbar ist. Das Verhalten ist somit aus unternehmerischer Sicht nachvollziehbar. Die Frage bleibt aber dennoch legitim, ob dieser Aufwand, der betrieben werden müsste, um nutzergenerierte Inhalte **vollständig** zu schützen, tatsächlich notwendig ist. Die IT-Sicherheit spricht bezüglich der Erreichung der Schutzziele auch lediglich davon, „dass Informationssysteme so zu gestalten [sind], dass die Gewährleistung

[24] beispielsweise durch eine konsequente Nutzung einer TLS/SSL-Verschlüsselung

[25] Bezeichnung für die neutrale Datenübermittlung im Internet. Sie bedeutet, dass Zugangsanbieter (access provider) Datenpakete von und an ihre Kunden unverändert und gleichberechtigt übertragen, unabhängig davon, woher diese stammen oder welche Anwendungen die Pakete generiert haben

[26] vgl. dazu die Zusammenstellung von Eric Butler zum Thema „Firesheep“, http://codebutler.com/firesheep-a-day-later

[27] vgl. erneut http://codebutler.com/firesheep-a-day-later

der Schutzziele in ausreichendem Maß erfolgt“. Wann ist jedoch ein Maß „ausreichend“? Ist private Kommunikation weniger schützenswert als geschäftliche oder behördliche?

Das Grundgesetz beantwortet zumindest für den politischen Raum der Bundesrepublik Deutschland diese Frage deutlich: Artikel 2 Absatz 1 des Grundgesetzes garantiert die **freie Entfaltung der Persönlichkeit** [Bun10]. Darauf aufbauend lässt sich der Begriff der **Privatsphäre** herleiten, der den nicht-öffentlichen Bereich bezeichnet, in dem ein Mensch unbehelligt von äußeren Einflüssen dieses Recht auf freie Entfaltung der Persönlichkeit wahrnimmt. Ein **Recht auf Privatsphäre** gilt als Menschenrecht und ist in allen modernen Demokratien verankert. Privatsphäre verlangt vertrauliche Kommunikation — auch beim Teilen im Web. Wenn nun allerdings ein Mensch über das Web vertraulich kommunizieren möchte, um sich frei mit anderen Menschen über private Dinge auszutauschen, so wird die Wahl seiner Kommunikationswerkzeuge aktuell begrenzt sein, sofern er sich überhaupt auf eine Anwendung einlassen kann, die seinem Grad an Gewährleistung von Vertraulichkeit entspricht. Es lassen sich schnell persönliche Inhalte ausdenken, die man auch nur beim geringsten Zweifel an der Vertraulichkeit der Kommunikation, *eher nicht* digital austauscht.

Auch — und vielleicht auch *gerade* — im privaten Kontext ist daher die Kommunikation aller Menschen aufgrund des Rechts auf Privatsphäre untereinander schützenswert. Es werden *nicht* besondere Nutzunsszenarien dafür vorausgesetzt, sondern vertrauliches Teilen sollte jedem Menschen zu jeder Zeit über das Web möglich sein. Im geschäftlichen Kontext ist der vertrauliche Austausch von Inhalten zusätzlich mit der Existenz von Betriebs- und Geschäftsgeheimnissen begründbar, deren Wissen allein im Unternehmen bleiben soll. In der behördlichen Kommunikation der Staatsorgane greifen wieder andere Anforderungen an die Vertraulichkeit von Kommunikation, z.B. die Geheimschutzordnung des Bundesrates, die vier Geheimhaltungsgrade von *Verschlusssachen* benennt, deren Befolgung helfen soll Gefährdungen von der Bundesrepublik und ihrer Länder abzuhalten [Bun86].

So erkennt man in beinahe jedem Kommunikationskontext verschiedene Anforderungen an Vertraulichkeit, an Integrität und auch an die Verfügbarkeit der eigenen Kommunikationsinhalte. Die Gewährleistung dieser Schutzziele mag daher nicht immer wirtschaftlich sein, sinnvoll ist sie **aus der Perspektive der Nutzer** ohne Zweifel.

1.3 Idee und Motivation

Die technischen Ursachen der vorgenannten Probleme sind nun im Einzelnen:

- Die zentrale Anwendungslogik, die sämtliche Benutzerdaten vermittelt,
- die zentrale Datenhaltung, in der Benutzerdaten persistent gespeichert werden, und schließlich
- die zentrale, übermächtige Betreiberinstanz, die diese beiden Systeme betreibt und kontrolliert.

Darauf aufbauend lässt sich eine erste Hypothese formulieren:

Hypothese 1 *Eine Web-Anwendung kann nur dann die Schutzziele der IT-Sicherheit vollständig gewährleisten, wenn sie nicht zentralistisch organisiert ist, sondern die Kontrolle über die Speicherung und Vermittlung der eigenen Daten allein bei deren Urhebern liegt.*

In der Folge wäre eine Demokratisierung der Systemkomponenten und die Auflösung der zentralen Betreiberinstanz denkbar. Der Betrieb einer Web-Anwendung könnte auf zahlreichen von den Nutzern der Anwendung kontrollierten Systemen stattfinden. Statt eines einzelnen übermächtigen Betreibers, der sämtliche Systeme der Anwendung bereitstellt und wartet, wäre die Anwendungslogik auf die Systeme der Benutzer verteilt, ebenso die Datenspeicherung. Es entstände als Architektur ein Geflecht aus dezentral organisierten *Peers*, die eigenverantwortlich mit *ihren* Inhalten umgingen, sie lokal vorhielten und mit anderen Peers teilten.

Diese Herangehensweise, die Architektur einer Anwendung hin zu einem *Peer-2-Peer*-Netzwerk (in der Literatur auch *Distributed Social Network* genannt) grundlegend aufzulösen, kann ***ein*** denkbarer erster Schritt in Richtung der Lösung des Zentralismusproblems sein, sofern die Benutzer dadurch eine hinreichende Kontrolle über ihre Inhalte zurückerlangen und von einer Gewährleistung der Schutzziele auszugehen ist. Doch genügt es allein auf technischer Ebene die Architektur einer Web-Anwendung durch eine alternative Architektur zu ersetzen, um das Ziel, Menschen innerhalb von Web-Anwendungen vertraulich Inhalte miteinander teilen zu lassen, diese Daten gegen unbefugten Zugriff abzusichern und ihre Integrität und Verfügbarkeit sicherzustellen, hinreichend zu lösen? Selbst wenn die alternative Architektur aus technischer Sicht umfassend und unangreifbar[28] wäre und allein aufgrund ihres Aufbaus sicherstellte, dass die Schutzziele vollständig gewährleistet

[28] dieser Begriff ist bewusst als Superlativ gewählt, auch wenn kein System in der Informatik als „unangreifbar“ bezeichnet werden sollte.

würden, woher sollten Benutzer das Vertrauen nehmen, dass diese Qualität der alternativen Architektur tatsächlich vorhanden ist, wenn sich möglicherweise eine auf der Architektur aufbauende Web-Anwendung nach außen hin (Präsentationsschicht) aus Interaktionssicht gleich verhält wie die zu ersetzende?

Technik ist komplex. Architekturen von Web-Anwendung sind komplex und können meist nur schematisch und stark abstrahiert veranschaulicht werden. Nachzuvollziehen wie Systeme im Web überhaupt miteinander kommunzieren und was dies für Implikationen auf die Daten der Benutzer hat, dies ist meist Experten vorbehalten, die sich mit entsprechendem Fachwissen technische Zusammenhänge erschließen und diese bewerten können.

Es lässt sich aus diesem Gedanken heraus eine zweite Hypothese formulieren, die die durch die Benutzer wahrnehmbare *Vertrauenswürdigkeit* einer Architektur adressiert.

Hypothese 2 *Die Bereitschaft eine Web-Anwendung zu nutzen, die auf einer alternativen Architektur fußt, geht nur dann in eine tatsächliche Nutzung über, wenn die Anwendung selbst eine* ***nachvollziehbare Gewährleistung der Schutzziele*** *seitens der Nutzer ermöglicht. Eine Architektur zum Teilen nutzergenerierter Inhalte im Web sollte aus sich heraus* ***vertrauenswürdig*** *sein, um Motivation für eine Nutzung zu schaffen.*

Eine Architektur zum Teilen nutzergenerierter Inhalte im Web zu entwickeln, die sowohl die Erreichung der Schutzziele der IT-Sicherheit gewährleistet, als auch diesem grundlegenden Gedanken Rechnung trägt, Vertrauenswürdigkeit als fundamentale Qualität zu verankern, und dabei die formulierten Hypothesen zu prüfen, dies soll Inhalt der weiteren Ausarbeitung sein.

Forschungsfrage *Existiert eine Architektur, auf der Web-Anwendungen aufsetzen können, die die Schutzziele der IT-Sicherheit Integrität, Verfügbarkeit und Vertraulichkeit im Web geteilter Inhalte gewährleisten kann und dabei durch seine Nutzer als vertrauenswürdig erkannt wird?*

1.4 Vorgehen

Eine *vertrauenswürdige Architektur zum Teilen nutzergenerierter Inhalte im Web* soll entwickelt werden. Eine Architektur kann als die technische Grundlage von Web-Anwendungen angesehen werden. Sie beschreibt fundamentale Komponenten einer

Web-Anwendung, die Beziehung und Kommunikation unter diesen und die Organisation von Daten innerhalb des Systems. Eine Architektur kann auch anwendungsübergreifend verstanden werden und ein Gesamtgefüge aus verschiedenen Diensten im Web beschreiben. Sie erfüllt in jedem Fall auf technischer Ebene grundlegende Anforderungen an die Verarbeitung und Speicherung von Nutzerdaten und sie berücksichtigt auch aus einer Benutzerperspektive funktionale und nicht-funktionale Anforderungen an die Auswahl der bereitgestellten Funktionen zum Teilen von Inhalten und schließlich genügt sie Anforderungen an die Nachvollziehbarkeit ihrer Vertrauenswürdigkeit. All diese Anforderungen und Eigenschaft der Architektur sollen in einem mehrstufigen *Design*-Prozess ermittelt, diskutiert und schließlich in einen Architekturentwurf überführt werden.

1.4.1 Vorgehensmodell: Goal-directed Design

Als Vorgehensmodell während des Prozesses soll ein *Goal-directed Design*-Ansatz nach Cooper, Reimann und Cronin (2003) Verwendung finden [CRC07]. Das Goal-directed Design beschreibt die *Design*-Aktivitäten in einem benutzerzentrierten Software-Entwicklungsprozess und ein Set aus Methoden, deren Durchführung die Entwicklung gebrauchstauglicher Software unterstützen sollen. Bei der Betrachtung eines laut der Autoren „vollständigen“ Entwicklungsprozesses (wie in Abbildung 1.3 dargestellt) ist das Goal-directed Design überwiegend in der zweiten Konzeptphase (*Design*) zu verankern und umfasst alle konzeptionellen Aktivitäten, die vor und schließlich auch parallel zur tatsächlichen Implementierung (*Build*) einer Software stattfinden.

Abbildung 1.3: Ein vollständiger Software-Entwicklungsprozess nach [CRC07]

Das Goal-directed Design stellt die Ziele (Goals) der Benutzer zukünftiger Systeme in den Fokus aller Aktivitäten, die sich grundlegend von den mit dem System zu erfüllenden Aufgaben (Tasks) unterscheiden und deren Erreichung laut den Autoren der eigentliche Schlüssel zu einem intendierten *Joy of Use* ist und so letztlich die Qualität einer Benutzungsschnittstelle bestimmt.

Im Gegensatz zu anderen benutzerzentrierten Vorgehensmodellen zur Entwicklung gebrauchstauglicher Software – wie zum Beispiel dem *Usability Engineering Lifecycle* nach Mayhew [May99] oder dem *Scenario-based Development* nach Rosson und Carroll

[RC01] –, liegt beim Goal-directed Design eine besondere Betonung auf der Umsetzung der Analyseergebnisse (*Research*) in ein visuelles Design (*Design Refinement*): Das Modell soll damit eine Brücke schlagen zwischen Anforderungsermittlung und der konkreten Gestaltung der Nutzungsschnittstelle. Die Autoren verbinden *Research* und *Design Refinement* durch die Phasen *Modeling* (Überführung der Erkenntnisse aus der Research-Phase in strukturierte Modelle), *Requirements Definition* (Formulierung von Anforderungen auf Basis der Modelle) und *Design Framework* (Ermittlung fundamentaler Architekturelemente) und lassen schließlich als Symbol für die Weiterentwicklung des Designs während der tatsächlichen Implementierung der Software eine *Support*-Phase folgen.

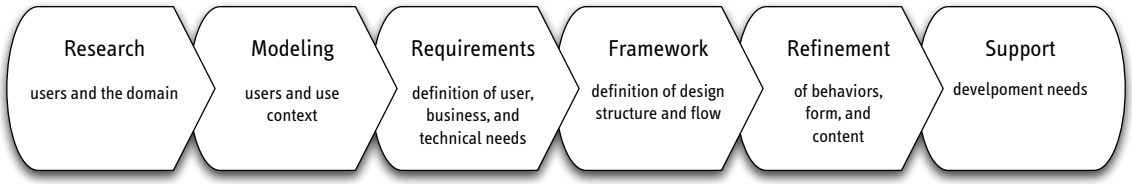

Abbildung 1.4: Der *Goal-directed Design*-Prozess nach Cooper und Reimann (2003) [CRC07]

Die inhaltlichen Schwerpunkte und Aktivitäten der einzelnen Phasen sind nun im Folgenden:

Research In der *Research*-Phase findet eine Analyse des Nutzungskontextes, der zukünftigen Nutzer, des Marktes, der technischen Rahmenbedingungen und konkurrierender Produkte statt. Primäre Ziele dieser frühen Phase sind durch Literaturrecherche, Benutzerbefragungen und Betrachtung ähnlicher Produkte ein tiefes Verständnis der Problemdomäne im interdisziplinären Entwicklerteam zu verankern, die Gruppen zukünftiger Nutzer und deren Ziele in Verbindung mit dem System zu ermitteln und ein gemeinsames Vokabular zwischen Benutzern und Entwicklern aufzubauen.

Modeling Die Ergebnisse der Research-Phase werden im Rahmen des *Modeling* in strukturierte Modelle überführt. Im Zuge einer Benutzermodellierung werden Personae angefertigt, die im weiteren Prozess die Partizipation der zukünftigen Benutzer darstellen und auch zur Prüfung neu ermittelter Anforderungen genutzt werden können. Darüber hinaus können beobachtete Arbeitsabläufe, Artefakte und Nutzungskontexte in Sequenzdiagramme oder schematische Darstellungen überführt werden. Hierzu finden vorwiegend Methoden des *Contextual Design* nach Beyer und Holtzblatt (1998) Verwendung [BH98].

Requirements Definition Anhand der Modelle wird im Rahmen der *Requirements Definition*-Phase ein weiterer Schritt in Richtung eines konkreten Architekturentwurfs unternommen: Eine ideale Benutzerinteraktion aus Sicht der Personae wird in Form von narrativen (aber strukturierten) Kontextszenarien formuliert, aus denen sich **Anforderungen** verschiedener Art an das System ergeben. Auf Basis mentaler Modelle der Benutzer und der hier ermittelten funktionalen, nicht-funktionalen oder auch datenbezogenen Anforderungen lässt sich dann im nächsten Schritt ein erstes Architekturfundament skizzieren.

Design Framework Als letzte Vorstufe vor dem konkreten Gestaltungsentwurf der Architektur soll im Rahmen des *Design Framework* nicht direkt auf Basis der ermittelten Anforderungen Detailarbeit geleistet werden, sondern es soll auf abstrakter *High-Level*-Ebene ein Blick auf das Fundament der Architektur gerichtet werden. Einem Top-Down-Ansatz folgendend werden zunächst grobe Skizzen von Technik und Benutzungsschnittstelle erarbeitet, die nach und nach an Detailgrad hinzu gewinnen — ohne dabei die Ziele der Benutzer aus den Augen zu verlieren.

Design Refinement Der konkrete Architekturentwurf wird schließlich in der *Design Refinement*-Phase erarbeitet. Konkrete technische Lösungen werden benannt und miteinander in Bezug gebracht. Auf visueller Ebene werden detailierte Interface Widgets der Benutzungsschnittstelle beschrieben, die genaue Darbietung von Informationen, die genutzte Sprache und der Aufbau des Storyboards.

Support Zuletzt benennen die Autoren eine Phase, welche den Design-Prozess in die Implementierung der Software übergehen lässt (vgl. dazu Abbildung 1.3). Sie hat den Zweck darauf hinzuweisen, dass sich auch *nach* der Erarbeitung eines (ersten) Architekturentwurfs während dessen Implementierung neue Anforderungen ergeben oder sich technische Rahmenbedingungen verändern können. In diesen Fällen ist der Architekturentwurf an die geänderten Umstände anzupassen — ein Prozess, der in einem interdisziplinären Team aus Designern und Entwicklern geschehen durchgeführt werden muss und ein „kollaboratives Design“ verlangt.

1.4.2 Bewertung des Goal-directed Design-Ansatzes im Kontext dieser Ausarbeitung

Auch wenn die Entwicklung einer *Architektur* zum Teilen von Inhalten im Web eher technik-getrieben und weniger visuell-gestalterisch erscheint, bleibt sie ein *Design-*

Prozess[29], wenn man statt der im Deutschen häufig mitschwingenden Konotation „Design ist vorwiegend visuelle Gestaltung“ den Begriff des Designs allgemeiner als die Gestaltung einer Architektur durch Auswahl und Komposition von Systemkomponenten versteht. Die Herleitung eines konkreten Architekturentwurfs lässt sich daher mit dem Goal-directed Design-Prozessmodell und den darin verwendeten Methoden sehr gut strukturieren und durchführen. Besonders geeignet für die Nutzung im Rahmen dieser Ausarbeitung erscheint das Goal-directed Design aufgrund seiner Aufforderung die Problemdomäne in der *Research*-Phase möglichst vielschichtig aufzuarbeiten, und der Möglichkeit diese Erkenntnisse im Anschluss benutzbar für den weiteren Entwicklungsprozess in Modellen abzubilden. Denn es ist einerseits ein tiefes technisches Verständnis der Domäne erforderlich, als auch die Bedeutung darin relevanter psychologischer und sozialwissenschaftlicher Phänomene, um schließlich Ideen entwickeln zu können, wie das Konzept der „Vertrauenswürdigkeit“ so in einer technischen Architektur verankert werden kann, dass es Benutzern über die Nutzungsschnittstelle erfahrbar wird und sie bei deren Zielerreichung unterstützt. Es ist anzunehmen, dass sich die Verankerung von Vertrauenswürdigkeit über alle Schichten einer Architektur erstreckt — beim technischen Fundament angefangen bishin zu einer visuell gestalteten Benutzungsschnittstelle einer darauf aufbauenden Web-Anwendung. Der *Goal-directed Design*-Prozess berücksichtigt all diese Zwischenstufen auf dem Weg zu einem konkreten Architekturentwurf.

[29] nicht zuletzt weil dies eben der englische Begriff für Architektur ist

2 Entwurf

Im Folgenden soll nun die stetige Annäherung an einen konkreten Entwurf einer *vertrauenswürdigen Architektur zum Teilen nutzergenerierter Inhalte im Web* stattfinden. Die strukturelle Grundlage bilden die ersten fünf Phasen des Goal-directed Design-Ansatzes: *Research* → *Modeling* → *Requirements Definition* → *Design Framework* → *Design Refinement*.

2.1 High-Level-Goals

Aus dem Kontext der Problemdomäne abgeleitet sollen dem Architekturprozess vier alles umschließende Ziele der zu entwerfenden Architektur vorangestellt werden. Dies sind:

- auf der zu entwerfenden Architektur basierend sollen Web-Anwendungen entwickelt werden oder bestehende Web-Anwendungen aufsetzen können, die das Teilen von nutzergenerierten Inhalten im Web ermöglichen
- sie soll die Schutzziele der IT-Sicherheit Vertraulichkeit, Integrität und Verfügbarkeit hinsichtlich der Nutzerinhalte hinreichend gewährleisten
- sie soll aus sich heraus *vertrauenswürdig* sein und dadurch Nutzungsmotivation bei potentiellen Nutzern schaffen
- sie soll sich in das aktuelle Web-Gefüge einbetten und die aktuellen Entwicklungen in der Problemdomäne berücksichtigen

2.2 Research

Im Rahmen der *Research*-Phase (dt. Recherche) wird die Problemdomäne umfassend analysiert. Dazu wird zunächst auf theoretischer Ebene das **Vokabular** der Domäne betrachtet, um eine begriffliche Basis für die anschließende Benutzerpartizipation zu schaffen. Im zweiten Teil der Recherche wird die potenzielle Benutzerzielgruppe der neuen Architektur ermittelt, es werden **Befragungen** ausgewählter Mitglieder durchgeführt und die gewonnenen Erkenntnisse dokumentiert. In einem dritten

Teil wird dann auf den Ergebnissen der Befragung aufbauend das **Ökosystem** der neu zu entwerfenden Architektur betrachtet: In welchen Kontext würde sich eine neue Architektur aktuell einbetten? Wie sehen existierende Teillösungen für erkannte Probleme aus und würde sich deren Adaption oder Nutzung eignen? Welche konkreten Systemen werden derzeit genutzt oder befinden sich in der Entwicklung, die erkannte Teilprobleme lösen und wie zufriedenstellend tun sie es?

2.2.1 Vokabular

Um die Befragung potenzieller Benutzer auf einer begrifflich fundierten Grundlage durchführen zu können, werden in diesem Abschnitt drei Begriffe des Vokabulars der Problemdomäne näher betrachtet: Was bedeutet **Teilen** im digitalen Kontext? Was ist **Vertrauen**? Wie entsteht es und findet man es auch im Web? Wann ist etwas **nachvollziehbar** und welche Qualitäten muss es dafür aufweisen?

Bei der Beantwortung dieser Fragen werden auch bereits erste, allerdings noch wenig konkrete Anforderungen an die zu entwickelnde Architektur formuliert.

2.2.1.1 Teilen

Teilen ist das gemeinsame Nutzen einer Ressource. Deren Ausprägung kann verschieden sein: Die Ressource kann materielles Gut sein (Geld, Äpfel, Maschinen, ...) und muss zum Teilen unter Menschen dann tatsächlich *aufgeteilt* oder *zerteilt* werden[1] oder sie kann immaterielles (Kultur-)Gut sein (Wissen, Erfahrung, Erinnerungen, Tradition, Schicksal, ...) und kann dann anderen *mitgeteilt* und dabei auch zeitgleich in vollem Umfang gemeinsam genutzt oder erlebt werden.

Dieses gängige Konzept des Teilens lässt sich allerdings nicht 1:1 in den Web-Kontext übertragen. Es gelten beim Teilen von digitalen Artefakten im Web besondere Regeln. Zwar mögen digitale Inhalte dem allgemeinem Verständnis nach eher als *materiell* empfunden werden[2], sie zu teilen geschieht auf andere Art und Weise: Weder werden sie selbst *auf-* und anderen *zugeteilt*, noch einfach *mitgeteilt*, vielmehr verbirgt sich hinter dem Prozess des Teilens im Web die *Duplizierung* eines digital repräsentierten Artefakts. Es wird dann diese digitale *Kopie* anderen zur Verfügung gestellt und damit schließlich dessen Nutzungsmöglichkeit *geteilt*. Während man also einen realen Apfel in n Stücke teilt und die Teilhaber dabei jene n Teile des Ursprungsapfels erhalten, die zusammengefügt widerum einen vollständigen Apfel ergeben, diesen

[1] oder es wird ihre Nutzungszeit aufgeteilt

[2] und in gewisser Hinsicht sind sie auch durch ihre Byte-Repräsentation physisch „materialisiert“

also unmittelbar referenzieren und teilidentisch mit ihm sind, ist die Referenz einer Kopie eines Apfel-Fotos auf dessen Original im Web nicht systembedingt, auch eine (Teil-)Identität mit dem Ursprungsartefakt existiert nicht. Es entsteht ein neues, wenn auch gleiches, Datum. Der Teilende gibt daher kein Teil seines Artefaktes exklusiv in andere Hände und verzichtet nicht auf die eigene Nutzung, sondern kann darüber nachwievor verfügen; und während sich daher materielle Dinge meist endlich oft (sinnvoll) teilen lassen oder sogar nur lokal in der Nutzung geteilt werden können, können Inhalte im Web beliebig oft dupliziert und damit mit beliebig vielen Menschen gleichzeitig geteilt werden.

Das Teilen eigener Inhalte im Web kann auch anynom geschehen, wie es in vielen *File-Sharing-Netzwerken* der Fall ist. Hier ist es Benutzern häufig unklar, *wer* einen Inhalt ursprünglich den Benutzern des Netzwerkes zur Verfügung gestellt und dadurch mit diesen geteilt hat. Diese Anonymitätsbeziehung existiert auch anders herum: Urhebern eines Inhaltes ist meist nicht mehr nachvollziehbar mit wem sie Inhalte tatsächlich geteilt haben, wenn das Netzwerk, in dem geteilt wird, offen gestaltet ist und sich jeder am Pool der geteilten Inhalte bedienen kann. Für hochgeladene Videos in File-Sharing-Netzwerken lassen sich zwar meist noch Zugriffsstatistiken ermitteln, jedoch keine vollständige Liste mit identifizierten Teilhabern, die sich eine Kopie dieses Videos erstellt haben — diese beidseitige Anonymität ist in diesem Fall allerdings ein essentieller Teil des Gesamtkonzeptes des Netzwerkes, da es den illegalen Austausch rechtlich geschützter und nicht zur Weitergabe freigegebener Daten vereinfacht.

Zusammenfassend lässt sich festhalten: Das Teilen von Inhalten im Web

- ist kein Zer- oder Aufteilen, sondern ein Duplizieren und zur Verfügung stellen,
- es geschieht für den Teilenden verzichtfrei,
- skaliert unbegrenzt (beliebig viele Inhalte können mit beliebig vielen Personen gleichzeitig geteilt werden),
- *kann* in beide Richtungen des Teilens anonym stattfinden (Teilender kennt Teilhaber nicht und/oder Teilhaber kennen Teilenden nicht),
- es kann auch jeder Teilhaber eines Inhaltes diesen erneut wieder mit anderen teilen und
- so zieht es dadurch zwangsweise einen Kontrollverlust des Teilenden über seine geteilten Inhalte nach sich.

Dass dieses im Gegensatz zur Offline-Welt komplexe und unglaublich mächtige Modell des Teilens von Inhalten im Web auch heutzutage immer noch zu Datenschutz-,

Privatsphären-, und Copyright-Problemen führt und dass dessen Implikationen für den verantwortungsbewussten Umgang mit vertraulichen Inhalten noch nicht vollständig verstanden scheinen, zeigen Fälle wie die jüngst als „Cablegate“ bezeichnete anonyme Veröffentlichung vertraulicher Depeschen von US-Botschaften durch die Enthüllungsplattform *WikiLeaks* [Wik10] [Sch10a]. 251.287 Dokumente, die von US-Botschaftern zum Teil als „vertraulich“, zum Teil auch als „geheim“ eingestuft wurden und Informationen über und Bewertungen von vielen internationalen Politikern beinhalten, gerieten aus dem Intranet der US-Diplomatie ins World Wide Web. Rund 250.000 Angestellte der US-Regierung hatten innerhalb des Intranets autorisierten Zugriff auf diese Daten. Sascha Lobo, deutscher Blogger und Buchautor, kommentiert den Vorfall in der ARD-Sendung *Anne Will* vom 28.11.2010 [Das10] als absehbare Folge einer unterschätzten Situation:

«Wir erleben hier den ersten Ausläufer einer neuen digitalen Gesellschaftsordnung, die es extrem schwierig bis fast unmöglich macht Daten geheim zu halten. [...] Es wird dutzende, hunderte, tausende Plattformen und Plattförmchen geben, wo solche delikaten Informationen veröffentlicht werden. [...] Daten werden in Zukunft weniger sicher sein, und zwar alle Daten. Und damit müssen wir umgehen. [...] Man kann sich nicht einbilden, dass in Zeiten des Internets 250.000 Menschen ein Geheimnis bewahren können. Es ist unmöglich. [...] Und genau diese Mechanik, dass jeder alles überall publizieren kann — und das auch noch anonym — die wird natürlich, je delikater ein Datum ist, umso interessanter für manche Leute.»

Das Fallbeispiel zeigt: Das einfache und unbegrenzte Teilen von Inhalten im Web kann zeitgleich das Leben in der modernen Gesellschaft durch freies Wissen und freie Informationen bereichern und eine weltweite multimediale Kommunikation unter den Menschen ermöglichen, es birgt allerdings auch Risiken und Gefahren beim Umgang mit vertraulichen Inhalten durch deren Unkontrollierbarkeit sobald sie in digitaler Form „in der Nähe des World Wide Webs“ vorliegen.

Folgerung Eine Architektur zum Teilen von Inhalten im Web, die die Schutzziele der IT-Sicherheit bezüglich der teilbaren Inhalte berücksichtigt, sollte das unbegrenzte, digitale Teilen ermöglichen, jedoch versuchen den Urhebern der Inhalte eine größtmögliche Kontrolle über ihre Inhalte zu bewahren. Die unautorisierte Weitergabe geteilter Inhalte sollte eingeschränkt werden.

2.2.1.2 Vertrauen

Der Vertrauensbegriff besitzt je nach Betrachtungsperspektive und -kontext verschiedene Konotationen, erfüllt verschiedene Funktionen und entfaltet verschiedene Wirkungen. Der Soziologe Niklas Luhmann betrachtet Vertrauen in erster Linie als Mittel, soziale Komplexität zu reduzieren und den alltäglichen Handlungsdruck beherrschbar zu gestalten.

«Vertrauen im weitesten Sinne eines Zutrauens zu eigenen Erwartungen ist ein elementarer Tatbestand des sozialen Lebens. Der Mensch hat zwar in vielen Situationen die Wahl, ob er in bestimmten Hinsichten Vertrauen schenken will oder nicht. Ohne jegliches Vertrauen aber könne er morgens sein Bett nicht verlassen. Unbestimmte Angst, lähmendes Entsetzen befielen ihn. [...] Solch eine unvermittelte Konfrontierung mit der äußersten Komplexität der Welt hält kein Mensch aus.»

— Niklas Luhmann [Luh00]

Unter Vertrauen wird folglich die Annahme verstanden, dass Entwicklungen einen erwarteten Verlauf nehmen oder sich die künftigen Handlungen von Personen oder Organisationen im Rahmen von gemeinsamen Werten oder moralischen Vorstellungen bewegen werden. Diese Annahme wirkt sich in der Gegenwart aus, ist allerdings auf künftige Ereignisse gerichtet. Dass man in diesem Kontext von „Erwartungen" spricht, lässt schlussfolgern, dass der tatsächliche Verlauf einer Entwicklung oder Handlung zum aktuellen Zeitpunkt unbekannt ist und aufgrund von persönlichen Vorbewertungen lediglich eingeschätzt werden kann. Vertrauen ist also auch ein Mittel, um mit Unwissenheit und Ungewissheit umzugehen. Gesellschaftsforscher Guido Möllering stellt dies anschaulich an einem langjährigen Slogan der Sportmarke *Nike* dar:

«Die Aufforderung „Just do it!" bringt den im Vertrauen entscheidenden, uns im Alltag zum Glück meist unbewussten Moment zum Ausdruck, in dem Akteure nicht immer weiter nach guten Gründen suchen, sondern ihre verbleibende Ungewissheit und Verwundbarkeit anderen gegenüber überwinden — sich selbst überwinden — und es einfach tun. Sie haben eine positive Erwartungshaltung erreicht und handeln, als ob negative Möglichkeiten nicht eintreten könnten und das Wohlverhalten anderer gewiss sei. Wenn sie vertrauen, sind sie nicht mutig oder verzweifelt oder lebensmüde, sondern davon überzeugt, dass man ihnen nicht schaden wird, obwohl sie dies nicht sicher wissen können und trotz des fälschlicherweise so genannten Restrisikos, das sie gar nicht ermessen können.»

— Guido Möllering [Mö10]

Das angesprochene Restrisiko dient als Symbol für die Eintrittswahrscheinlichkeit für den aus Sicht des Vertrauenden negativen Verlauf einer Entwicklung. Doch im Gegensatz zur *Hoffnung*, dass eine Situation einen bestimmten negativen Verlauf nicht nimmt, unterscheidet sich das Vertrauen als positive Erwartungshaltung darin, dass der Vertrauende Handlungsalternativen besitzt, das Vertrauen gegenüber einer Situation bewusst aufbringt und ihr nicht schutzlos ausgeliefert ist.

Als wesentliche Merkmale von Vertrauen im soziologischen Sinne sollen aus dieser Betrachtung festgehalten werden:

- Die Existenz einer (sozialen) Beziehung zwischen einem Vertrauensnehmer und einem Vertrauensgeber,
- der Aspekt der Ungewissheit hinsichtlich künftiger Entwicklungen,
- das Vorhandensein eines Risikos für den Vertrauensgeber,
- das Vorhandensein von Handlungsalternativen auf Seiten des Vertrauensgebers,
- die dann jedoch anschließende mangelnde Beeinflussung des Schicksals und
- die Zeitperspektive (Vertrauen ist auf die Zukunft ausgerichtet), vgl. [Mö10] [Pet96].

In der wissenschaftlichen Literatur stößt man auf zahlreiche differenziertere Verwendungen und Deutungen von Vertrauen — in der Wirtschaftswissenschaft und dem Marketing spricht man beispielsweise von *Preis-* oder *Markenvertrauen*, in der Politikwissenschaft vom *Institutionenvertrauen*, welches die Bevölkerung in die Fähigkeit von Institutionen haben kann, oder in der Entwicklungspsychologie vom Konzept des *Urvertrauens* — im Kontext dieser Ausarbeitung erscheint die hier angesprochene soziologische Betrachtung des Vertrauensbegriffs jedoch am sinnvollsten, da sie unmittelbar verbunden ist mit einer zentralen Fagestellung der Ausarbeitung: Wenn beim Teilen von Inhalten im Web Vertrauensbeziehungen zwischen der Architektur und dessen Benutzern existieren sollen, um trotz Ungewissheit eine Handlungsfähigkeit des Benutzers herzustellen, wie entsteht dann eine positive Erwartungshaltung gegenüber den Intentionen und dem Verhalten anderer Personen oder anderer Systeme und welche Merkmale muss eine Person oder ein System aufweisen, um als *vertrauenswürdig* zu gelten?

Luhmann beschreibt in diesem Zusammenhang Vertrauen als einen Akt der Selbstdarstellung:

«Vertrauen ist dann die generalisierte Erwartung, dass der andere seine Freiheit, das unheimliche Potential seiner Handlungsmöglichkeiten, im Sinne seiner Persönlichkeit handhaben wird —- oder genauer, im Sinne der Persönlichkeit, die er als die seine dargestellt und sichtbar gemacht hat. Vertrauenswürdig ist, wer bei dem bleibt, was er bewusst oder unbewusst über sich selbst sichtbar gemacht hat.»

— Niklas Luhmann [Luh00]

Diesem Verständnis nach kann Vertrauenswürdigkeit nicht zu Beginn einer sozialen Beziehung existieren, sondern die Vertrauenswürdigkeit einer Person ist immer das Ergebnis sozialer Interaktion und deren individueller Bewertung. Erst wenn man Gelegenheit hatte sich ein Bild der Persönlichkeit eines anderen Menschen zu machen, kann eine Bewertung stattfinden, ob aktuelle Handlungen einer Person mit deren Persönlichkeit konform sind, also zu erwarten waren. Auf dieser Bewertung basierend kann dann ein Vertrauen in künftige Handlungen (als Erwartungshaltung) entwickelt werden und die subjektive Vertrauenswürdigkeit der Person steigen.

Dies kann für eine Abbildung auf eine technische Architektur bedeuten, dass die Architektur seinen Benutzern die Möglichkeit zur Erkenntnis der eigenen Funktionsweise (als Abbild der *Persönlichkeit*) geben sollte. Sie sollte in den grundlegenden Merkmalen nachvollzogen werden können und es sollte dann während der Benutzung der Architektur (als Abbild der *sozialen Interaktion*) überprüfbar sein, ob die vorher kommunizierte, vom Benutzer nachvollzogene Funktionsweise tatsächlich stattgefunden hat.

Folgerung Eine Architektur zum Teilen von Inhalten im Web, die als *vertrauenswürdig* empfunden werden soll, sollte ihren Aufbau und ihre Funktionsweise offen und nachvollziehbar kommunizieren und Möglichkeiten zur Überprüfung anbieten.

2.2.1.3 Nachvollziehbarkeit

Der vorangegangene Abschnitt wirft die Frage auf, wann der Aufbau und die Funktionsweise einer Architektur für ihre Benutzer *nachvollziehbar* ist.

Nachvollziehbar bedeutet im Allgemeinen, dass sich ein Vorgehen erschließt und steht in direktem Zusammenhang mit der Verständlichkeit eines Vorgehens. Etwas nachzuvollziehen kann dabei Fachkenntnis voraussetzen, besitzen jedoch zwei Personen den gleichen Kenntnisstand, sollte sich ihnen das Vorgehen auf die gleiche Art und Weise erschließen. Eine ideale Nachvollziehbarkeit beansprucht jene scheinbare

Objektivität. Sie ist *scheinbar* da durch die beteiligten Subjekte lediglich *Intersubjektivität* — das gleiche Verständnis eines Sachverhaltes für mehrere Betrachter — erreicht werden kann.

Die englische Übersetzung von Nachvollziehbarkeit *traceability* legt eine zweite deutsche Bedeutung nahe, die in Bezug auf die Nachvollziehbarkeit der Vorgänge innerhalb einer technischen Architketur verwendbar erscheint: Zurückverfolgbarkeit. Etwas nachzuvollziehen bedeutet auch alle Schritte eines Vorgehens zurückverfolgen zu können bis zu seinem Ursprung. Es entsteht hier das Bild einer Prozesskette, deren Bestandteile und die Zusammehänge unter diesen für den Betrachter verständlich sein müssen.

Folgerung Eine Architektur zum Teilen von Inhalten im Web, deren Alleinstellungsmerkmal „kontrolliertes Teilen“ für ihre Benutzer nachvollziehbar sein soll, sollte alle Schritte des komplexen Prozesses des Teilens deutlich an die Benutzer kommunizieren können. Denn Nachvollziehbarkeit bedingt das Vorhandensein aller notwendigen Informationen zu einem Vorgehen.

2.2.1.4 Nutzergenerierte Inhalte

Wenn moderne Web-Anwendungen bildlich gesprochen den „Motor“ des *Web 2.0* bilen, so sind die so genannten *nutzergenerierten Inhalte* (engl. *User-generated content*) der Treibstoff, der durch sie durchgepumpt wird. Der Übergang vom *read Web* zum *read/write Web* war auch der begriffliche Ursprung für die nutzergenerierten Inhalte des Web. Unter diesen werden landläufig im Web durch Nutzer veröffentlichte Daten eingeordnet, doch ist dieser Begriff differenzierter zu betrachten.

Die Organisation für wirtschaftliche Zusammenarbeit und Entwicklung (OECD) stellt in einem Report über das partizipative Web drei Anforderungen auf, die nutzergenerierte Inhalte zu erfüllen haben [fECoD07]:

- **Sie müssen öffentlich oder (teilöffentlich[3]) sein.** Dies grenzt nutzergenerierte Inhalte von E-Mails oder Instant Messaging ab als zwei Kommunikationsmittel des Internets, über die primär nicht-öffentliche, private Kommunikation erfolgt (oder Kommunikation erfolgt, die nicht für eine Veröffentlichung bestimmt war).
- **Es muss ein kreativer Aufwand ersichtlich sein.** Dies grenzt sonstige Daten,

[3] teilöffentlich bedeutet einer Untermenge der Öffentlichkeit zugänglich

die durch eine Interaktion von Nutzern mit dem Web entstehen von nutzergenerierten Inhalten ab, z.B. Nutzungsstatistiken oder Log-Dateien.

- **Sie dürfen nicht in einem professionellen Kontext entstanden sein.** Dies soll offenbar besonders die amateurhafte Qualität von nutzergenerierten Inhalten in den Vordergrund stellen, jedoch scheint diese Anforderung sehr diffus: Die Grenzen von professionellen und nicht-professionellen Produktionskontexten sind mit fortschreitender Technik auch im *Consumer*-Bereich kaum mehr erkennbar. Auch ist für die Konsumenten nutzergenerierter Inhalte meist nicht offensichtlich, ob etwas in einem professionellen oder nicht-professionellen Kontext entstanden ist.

Hagemann und Vossen (2009) kritisieren in ihrer Arbeit zur Kategorisierung nutzergenerierter Inhalte die Kriterien der OECD als nicht umfassend genug [HV09]. Die Betrachtung nutzergenerierter Inhalte könne sich nicht nur auf die klassischen Inhaltstypen von Wikis, Weblogs, Foren und Social Networks beziehen, sondern müsse auch Metadaten und Strukturen umfassen. Sie unterscheiden vier Typen nutzergenerierter Inhalte im Web:

- **Nutzergenerierte Inhalte im engeren Sinne** sind die klassischen Inhalte aus Wikis, Weblogs, Foren, Social Networks und anderen Plattformen, die Nutzern die Möglichkeit geben eigene Inhalte zu erzeugen. Die hauptsächliche Interaktion andere Nutzer mit diesen Inhalten sei vorwiegend auf den Konsum beschränkt. Nutzergenerierte Inhalte im engeren Sinne werden primär durch andere gelesen, angeschaut oder angehört.

- **Nutzergenerierte Strukturen** hingegen sind davon abzugrenzen, auch wenn sie in den gleichen Web-Diensten verankert sind. Die durch Tagging oder Verlinkung durch Nutzer geschaffenen Beziehungen zwischen Inhaltsobjekten sind zwar elementarer Bestandteil vieler Web-Anwendungen, jedoch ist deren Hauptzweck nicht dass sie von anderen gelesen werden (z.B. die URL eines Hyperlinks), sondern dass sie andere Inhalte referenzieren oder zueinander in Beziehung setzen.

- Als **nutzergenerierte komplexe Objekte** bezeichnen die Autoren Inhalte, die weder den Konsum, noch eine Strukturschaffung als Erstellungszweck besitzen. Als Beispiel wird die Firefox-Erweiterung *Sxipper*[4] angeführt. Diese hilft einem Benutzer dabei, Registrierungsformulare von Web-Anwendungen automatisch mit lokal abgespeicherten Stammdaten des Benutzers auszufüllen. In vielen Fällen kann das Mapping der lokalen Daten auf die Formularfelder aufgrund eines Algorithmus von Sxipper geschehen. Sollte dies jedoch einmal

[4] http://www.sxipper.com/

nicht funktionieren, so hat ein Benutzer die Möglichkeit das Mapping zwischen Stammdaten und Formularfeldern selbst durchzuführen und dieses Wissen als *nutzergenerierten Inhalt* den anderen Benutzern von Sxipper zur Verfügung zu stellen, so dass bei diesen auf das Mapping des einen Benutzers zurückgegriffen werden kann, sollten sie auch einmal das gleiche Formular ausfüllen. Zweifelsohne handelt es sich hier um einen nutzergenerierten Inhalt im Web, der sich jedoch in die ersten beiden Kategorien nicht einordnen lässt.

- **Nutzergenerierte Funktionalität** stellt aus Sicht der Autoren die vierte Gruppe von nutzergenerierten Inhalten im Web dar: Gemeint sind durch andere Benutzer *anwendbare* Inhalte, z.B. von Nutzern erstellte Suchindizes, Inhaltsfilter oder Mashups. In diesen Fällen werden nicht primär Inhalte erzeugt, sondern komplett neue Funktionalitäten, die widerum von anderen genutzt werden kann. Es bleibt jedoch eine Form von nutzergeneriertem Inhalt.

Nach Hagemann und Vossen sind nutzergenerierte Inhalte demnach mindestens anhand ihres Erstellungszwecks und ihrer Nutzungsmöglichkeit zu differenzieren. Zusätzlich ließe sich auch die Bedeutung des Inhaltes für seine umgebene Plattform beschreiben: Während z.B. ein Video für sich alleine Qualität[5] beanspruchen kann, so kann beim Tagging die Qualität eines Tags nur in der Gesamtheit aller Tags auf Plattformebene bestimmt werden.

Folgerung Beim Teilen nutzergenerierter Inhalte ist im Rahmen dieser Ausarbeitung das Verständnis der nutzergenerierten Inhalte im engeren Sinne vorzuziehen. Es sollen konkret diese durch andere konsumierbaren Inhalte in den Vordergrund gestellt werden, da bei Ihnen überhaupt eine Vertraulichkeit sowie eine Gefährdung der Integrität und Verfügbarkeit angenommen werden kann. Bei nutzergenerierten Strukturen, wie auch bei komplexen Objekten oder Funktionalitäten ist eine Erstellung zur anschließenden Veröffentlichung anzunehmen. Die Abgrenzung zu E-Mail oder Instant Messaging der OECD durch das Kriterium der Veröffentlichung im Web oder Teilveröffentlichung im begrenzten Benutzerkreis soll Geltung finden. Eine Architektur zum Teilen nutzergenerierter Inhalte im Web sollte daher primär die Multicast-Kommunikation 1:n adressieren.

2.2.2 Benutzerbefragung

Eine zentrale Komponente des Goal-directed Designs ist die Benutzer-Partizipation im Design-Prozess. Als benutzerzentrierter Ansatz soll dieser genau deren Ziele

[5] „Qualität“ im Sinne der Norm DIN EN ISO 9000:2005 [fS05]

in den Mittelpunkt stellen als Grundlage, um daraus im weiteren Verlauf Anforderungen zu ermitteln. Der folgende Abschnitt beschreibt die umfassende Integration potentieller, künftiger Benutzer in die Entwicklung der neuen Architektur. Diese beginnt bei der initialen Frage nach der **Zielgruppe** — wer sind potenzielle Benutzer, die es einzubeziehen gilt? — und unter Verwendung welcher **Methode** eine Analyse deren Ziele möglich ist. Die Durchführung individueller Benutzerbefragungen wird dokumentiert und deren **Ergebnisse** werden aufbereitet dargestellt.

Den Rahmen der Benutzerbefragungen bildet dabei die Annahme des Goal-directed Designs, dass qualitative Recherche über quantitativer steht: Aus empirischen Daten lassen sich nur schwer individuelle Ziele potentieller Benutzer ermitteln. Denn sie abstrahieren stets auf die komplexen Situationen, die Benutzer mit dem System erleben werden und können so zwar gut Fragen beantworten, die nach dem Muster „Wie viele Benutzer haben zu Tatsache X die Meinung Y?“ aufgebaut sind, nicht aber bei der vorgeschalteten Ermittlung der Tatsachen und Meinungen überhaupt helfen.

2.2.2.1 Zielgruppe

Die Frage nach der Zielgruppe potentieller, künftiger Benutzer eines Systems steht stets am Anfang einer Benutzerpartizipation im Entwicklungsprozess. Deren Beantwortung ist dabei gleichermaßen komplex wie folgeträchtig: Wird die Zielgruppe zu Beginn nur unscharf definiert oder werden ihr Benutzer zugeordnet, die später doch nicht mit dem System in Beziehung stehen werden, dann kann dies die Konsequenz haben, dass auf falsche Anforderungen geschlossen wird oder wesentliche Anforderungen übersehen werden. Im Folgenden finden daher verschiedene Herangehensweisen Verwendung, um eine Gruppe aus Stakeholdern zu identifizieren.

Exkurs: Stakeholder Dix et al. definieren in ihrem Standardwerk zur benutzerzentrierten Softwareentwicklung *Human-Computer-Interaction* (2003) die Stakeholder eines Systems als die

«Personen oder Personengruppen, die in irgendeiner Weise an dem Erfolg oder Misserfolg eines Systems und dessen Entwicklungsprozesses Interesse haben oder betroffen sind.»
vgl. [DFAB03]

Es werden dabei vier Klassen an Stakeholdern unterschieden, da nicht alle gleichberechtigt mit dem System in Beziehung stehen. Man unterscheidet zwischen

- einem **primären Stakeholder**, der in direkter Interaktion mit einem System steht, weil er es unmittelbar benutzt,
- einem **sekundären Stakeholder**, der das System nicht regelmäßig oder nicht direkt, sondern vielleicht nur durch Zwischenpersonen (primäre Stakeholder) nutzt, indem er z.B. Reports erhält, die mit dem System angefertigt wurden,
- einem **tertiären Stakeholder**, der weder als primärer oder als sekundärer Stakeholder eingeordnet werden kann, jedoch trotzdem vom Erfolg oder Misserfolg des Systems betroffen wäre. Dies können z.B. Entscheidungsträger in Unternehmen sein, die die Einführung des Systems beschlossen haben, jedoch weder direkt noch indirekt damit interagieren.
- Schließlich können als **unterstützende Stakeholder** die Personen bezeichnet werden, die in die Entwicklung, das Design und die Wartung des Systems integriert sind.

Identifizierung künftiger Benutzer

Die potentielle Gruppe zukünftiger Benutzer einer *vertrauenswürdigen Architektur zum Teilen von nutzergenerierten Inhalten im Web* kann allein durch die Tatsache der Nutzung der Architektur *im* Web extrem groß ausfallen. Denn wenn *eine* Annahme ist, dass zukünftige Nutzer bereits jetzt einen Zugang zum Web besitzen, dann kann jeder Web-Nutzer als potentieller Nutzer angesehen werden. Diese erste Annahme erleichtert die Anforderungsermittlung *nicht*, da man praktisch alle Individuen des Webs zu ihrer Beziehung zur neuen Architektur und dessen Problemdomäne befragen müsste. Die Gruppe der potentiellen Nutzer wird aus diesem Grund im Folgenden anhand der zentralen Fragen der „Persona Hypothese“ des Goal-directed Designs differenziert:

- Welche unterschiedlichen **Typen** von Personen könnten künftige Nutzer sein?
- Welche **Rolle** nehmen diese in der Interaktion mit dem System ein?
- In welchen **Dimensionen** könnten sich deren Verhalten und Anforderungen unterscheiden?
- Welche **Ausprägungen** besitzt jede dieser Dimensionen?

Die Beantwortung der Fragen geschieht in einigen sich anschließenden, multiperspektivischen Betrachtungen der Zielgruppe. Neue Annahmen über und Eigenschaften von potentiellen Nutzern werden aufgestellt und in einem Überblick zusammengefasst. Aus diesen „Anforderungen an potentielle Nutzer“ lassen sich dann Inter-

viewpartner bestimmen, die hinsichtlich ihrer Einstellungen, Motivation, Erfahrung, Sichtweisen und schließlich ihrer Ziele in Bezug auf die zu entwerfende Architektur befragt werden.

Differenzierung nach Nutzungsrolle Mit einer Architektur zum Teilen eigener Inhalte interagieren Personen in mehreren Rollen. Eine Rolle ist die des Produzenten/Teilenden, der Inhalte erstellt und diese anderen zur Verfügung stellt. Eine andere Rolle ist die des Konsumenten, der keine eigenen Inhalte erstellt, sondern nur fremde Inhalte konsumiert. Eine gleichzeitige Ausfüllung beider Rollen durch eine Person erscheint in einem lebendigen Teilungskontext als die wahrscheinliche Verteilung der Rollen. Nichtsdestotrotz sind es valide Nutzungsszenarien nur eine der beiden Rollen auszufüllen. Darüber hinaus können weitere Rollen existieren, die nicht mehr unmittelbar zu den primären Stakeholdern gezählt werden, z.B. Administratoren, die das System technisch betreuen werden.

Differenzierung nach Nutzungsmotivation Offensichtlich lassen sich zwei grundlegende Arten des *Teilens* beobachten: das private Teilen von Inhalten im Familien- oder Freundeskreis und das berufliche *Teilen* im Kollegen- oder Geschäftspartnerkreis. Es sind hier unterschiedliche Motivationen und Ziele anzunehmen. Während beim privaten Austausch von Dingen vorrangig Freude und die soziale Interaktion im Vordergrund stehen mögen, orientiert sich die geschäftliche Art des Teilens wahrscheinlich eher an den Zielen einer Unternehmung und ist darüber hinaus möglicherweise weniger selbstbestimmt als durch einen Prozess oder Vorgesetzten erzwungen. Um in diesen Punkten gesichtere Aussagen treffen zu können, sollen als Stakeholder daher Personen in Betracht gezogen werden, die sich möglichst überlappungsfrei diesen beiden Kontexten zuordnen lassen.

Differenzierung nach Nutzungskontext Die Zugangspunkte zum Web sind nicht begrenzt auf stationäre Desktop-Computer. Gerade in der jüngeren Vergangenheit gewinnt die mobile Nutzung des Webs durch die Verbreitung von Smartphones und sonstiger portabler internetfähiger Geräte an Popularität. Der mobile Nutzungskontext erlaubt zudem reichere Möglichkeiten des Teilens: Zum Beispiel können mit einem Smartphone aufgenommene Fotos direkt aus der Situation heraus geteilt werden. Es ist anzunehmen, dass eigene Inhalte nicht ausschließlich im Desktop/Browser-Kontext geteilt werden möchten, sondern auch über diverse Browser-ähnliche und Browser-unähnliche Client-Anwendungen, die auch auf mobilen Endgeräten installiert sein können. Auf die Gruppe potentieller Interviewkandidaten bezogen ließe sich schlussfolgern, dass diese auch Personen berücksichtigen sollte, die aufgrund

ihres Lebensstils oder ihrer Arbeit viel auf Reisen sind und so das Web zwangsweise primär mobil nutzen und aus dieser intensiven Erfahrung berichten können.

Differenzierung nach aktueller Nutzung Es ist anzunehmen, dass potentielle Nutzer sowohl aus dem Kreise der Personen stammen, die aktuell bereits ähnliche Software zum Teilen von Inhalten nutzen (z.B. Wikis, Weblogs oder Social Networks), allerdings auch aus der Gruppe an Personen, die diese Art von Software aktuell bewusst nicht nutzen (da sie womöglich Anforderungen, z.B. hinsichtlich der Vertraulichkeit, noch nicht erfüllt) oder denen sie gänzlich unbekannt ist. Stakeholder sollten sich auch in dieser Dimension unterscheiden, um auf die Nutzungsmotivationen und Nutzungsvoraussetzungen Rückschlüsse zu gewinnen.

Differenzierung nach Domänenwissen Das Teilen von Inhalten im Web ist ein komplexer Vorgang — sowohl im Prozessverlauf, als auch in der technischen Realisierung. Um Informationen über die Nachvollziehbarkeit dieses Vorgangs zu erhalten, sollen sich Stakeholder auch in ihrem Domänenwissen unterscheiden: Es sollen Personen in den Entwicklungsprozess einbezogen werden, die fundiertes technisches und konzeptionelles Verständnis des Webs besitzen, als auch Personen, denen die technische Grundlage völlig fremd ist und die kein Fachwissen aus der Informatik besitzen. Es ist anzunehmen, dass die Anforderungen an die Kommunikation der einzelnen Schritte des Teilungsprozesses zwischen diesen Gruppen unterschiedlich sein werden.

Differenzierung nach Inhalten Aufbauend auf dem Verständnis nutzergenerierter Inhalte aus Abschnitt 2.2.1.4 können diese unterschiedliche Qualitäten aufweisen: Sie können textuell oder multimedial sein, Hypertext oder *Rich Text*, endliche oder kontinuierliche Datenstreams, von den Dateigrößen besonders groß oder eher klein. Auch in diesem Punkt, der Art des präferierten Inhaltes, sollten sich die einbezogenen Nutzer unterscheiden.

Differenzierung nach bevorzugter Öffentlichkeit Die OECD beschreibt als eine Eigenschaft nutzergenerierter Inhalte deren Öffentlichkeit oder Teilöffentlichkeit. Es ist sicherlich aufschlussreich in welchen Dimensionen sich die Nutzergruppen zusätzlich unterscheiden, die bevorzugt öffentlich oder bevorzugt teilöffentlich, d.h. innerhalb einer begrenzten Nutzergruppe, Inhalte miteinander teilen. Zwar ist Vertraulichkeit eines Inhaltes bei einer vollständigen Veröffentlichung des Inhaltes nicht zu erreichen, aber dann auch nicht das Ziel. Es ist vorstellbar, dass Benutzer

über eine vertrauenswürdige Architektur Inhalte auch öffentlich teilen, wenn sie gleichzeitig die übrigen Qualitäten der Architektur, nämlich die Verfügbarkeit und Integrität der Inhalte, gesichert sehen möchten.

Differenzierung nach dem Verständnis „hinreichender Sicherheit“ Eine absolute Systemsicherheit herzustellen ist in der Informatik ein komplexes bis unmögliches Unterfangen. Dass nämlich *kein* computerbasiertes System von sich behaupten sollte, es sei „sicher“ zeigen die Aktivitäten des Chaos Computer Clubs der letzten 30 Jahre: Angefangen beim legendären Hack des BTX-Systems der Deutschen Post 1984 [Jou84], dem KGB-Hack wenige Jahre später [ste07], dem GSM-Hack Ende der Neunzigerjahre [ho02] bishin zu den Enthüllungen um die Unsicherheit von NEDAP-Wahlcomputern in der jüngeren Vergangenheit [Bun09], sind zahlreiche Systeme ihrer angenommenen „absoluten Sicherheit“ beraubt worden, die vorher teilweise sogar von offizieller Stelle bescheinigt wurde (im Falle des Wahlcomputer-Hacks hatte die Physikalisch-Technische Bundesanstalt die Sicherheit der Geräte geprüft).

Ein computerbasiertes System kann daher für jeden individuellen Anwendungsfall nur eine *hinreichende Sicherheit* bieten. Es obliegt dem Nutzer die Sicherheit eines Systems dahingehend zu bewerten, ob sie für den angestrebten Anwendungsfall hinreichend sicher ist. Auch die hier zu entwerfende Architektur wird nur eine hinreichende Sicherheit besitzen. Es wird Anwendungsfälle geben, die weitere Sicherheitsanforderungen an ein System stellen, und deren Unterstützung diese Architektur nicht leisten kann. Die in Abschnitt 1.2.5 erwähnte Geheimschutzordnung des Bundesrates beschreibt z.B. vier Geheimhaltungsgrade von Dokumenten. Die technischen Maßnahmen die zur Absicherung des Austausches von „streng geheimen“ Dokumenten getroffen werden müssen, mögen sich von denen unterscheiden, die ein Privatanwender für notwendig hält. Diesem kann bereits die Gewissheit genügen, dass seine vertrauliche Kommunikation zwar theoretisch mit endlichem Aufwand gehackt werden könnte, jedoch dieser Aufwand — seiner subjektiven Beurteilung nach — in keinem Verhältnis zur tatsächlichen Vertraulichkeit seiner Kommunikationsinhalte steht, er die Durchführung eines Hacks also für unwahrscheinlich hält und sich so keiner Gefahr ausgesetzt sieht.

Auch hinsichtlich dieses Verständnisses von *hinreichender Sicherheit* werden sich die Anforderungen unterschiedlicher künftiger Benutzer unterscheiden.

Differenzierung nach Wertepräferenzen Der Psychologe Dr. Peter Kruse hat sich mit seiner Beobachtung und der Frage beschäftigt, warum das Internet in öffentlichen Diskussionen die Gesellschaft polarisiert, und er formuliert die beobachtete

Schärfe des Disputs in einem Vortrag auf der Konferenz re:publica 2010 in Berlin als einen „Indikator für die Existenz unzureichend reflektierter Wertedifferenzen“ in der Gesellschaft [Kru10]. Um diese oft kulturell verkankerten Wertewelten zu erkennen, befragte er 191 so genannte *Heavy User* des Internets zu ihren Wertepräferenzen. Die Gruppe der Befragten wurde dabei demographisch bewusst breit gewählt und umfasst alle Altersklassen zwischen 14 und 50+ Jahren mit Haushaltsnettoeinkommen zwischen <1.5000 Euro und 3.000+ Euro und sie besitzt auch hinsichtlich der Geschlechter eine Gleichverteilung. Allerdings stellt sie **nicht** den Durchschnitt der deutschen Bevölkerung dar, sondern repräsentiert allein die Personen, die überdurchschnittlich aktiv im Internet sind. Zur Erhebung der Daten wählten die Befragten aus zahlreichen Paarvergleichen ihre präferierte Aussage. So entstand pro Teilnehmer eine multidimensionale Matrix seiner individuellen Wertepräferenzen. Über eine anschließende Mustererkennung konnte Kruse Cluster aus Teilnehmern bilden, Korrelationen zwischen Werteaussagen herstellen und dabei zwei grundlegend disjunkte Cluster an Personen differenzieren, die zu zahlreichen Themengebieten komplementäre Wertevorstellungen besitzen[6]. Diese beobachteten Wertedifferenzen zwischen den Clustern existieren in allen Altersklassen.

Die Wertepräferenz der ersten Gruppe beschreibt Kruse zusammenfassend als „Verlässliche Information und Beziehungen“. Demgegenüber steht die zweite Gruppe mit der Präferenz „Dynamik verstehen und aktiv mitgestalten“. Tabelle 2.1 stellt die Wertedifferenzen zwischen diesen Gruppen in einer Übersicht dar und zeigt außerdem eine Auswahl dominanter Themen innerhalb der Gruppen. Interessant ist die weitere Erkenntnis, dass beide Gruppen das Web zwar als ein dynamisches Konstrukt erkennen und sein „Wesen“ laut Kruse richtig einschätzen und akzeptieren, sich die eigene Wertepräferenz dabei jedoch nur bei der zweiten Gruppe mit den Eigenschaften des Webs deckt. Sie wollen die Virtualität genießen, Dynamik mitgestalten, Freunde im Social Web haben und persönliche Kontakte über das Internet pflegen, während diese Arten der sozialen Interaktion den Vertretern der ersten Gruppe völlig fremd sind. Kruse bezeichnet die Vertreter der ersten Gruppe als *Digital Visitors*, die zwar das Web akzeptieren und bereit sind es benutzen, jedoch nicht seine Werte vertreten, im Gegensatz zu den Vertretern der zweiten Gruppe, den *Digital Residents*, die Kontake im Internet als „Freundschaft“ und das Agieren im Web als „Leben“ bezeichnen können. Eine Übersicht zu den Wertewelten der beiden Gruppen ist dem Anhang A.1 zu entnehmen.

Kruses Ziel war es, sich unkonstruktive Diskussionen der Öffentlichkeit über das Web zu erklären. Konstruktiv könne man nur auf Faktenebene diskutieren. Wenn aber selbst bei den Benutzern des Webs, die sich am häufigsten und intensivsten

[6] und innerhalb eines Clusters die gleichen Werte präferieren

	Gruppe 1: „Digital Visitors“	**Gruppe 2: „Digital Residents“**
Wertepräferenz	Verlässliche Information und Beziehungen	Dynamik verstehen und aktiv mitgestalten
In der Gruppe dominante Themen	Entschleunigung, fundierte Analyse, geprüfte Richtigkeit, echte Begegnung, geschützter Raum, Qualitätssicherung, Vertrauen aufbauen, stabile Beziehung, Einfühlungsvermögen, Verlässlichkeit, Datenschutz	Muster erkennen, Dynamik erleben, unterhalten werden, Virtualität genießen, von überall ins Netz, Long Tail nutzen, Dynamik mitgestalten, inspirierende Vielfalt, Authentizität, Reputation pflegen
Perspektive	erkennt die Eigenschaften und Werte des Webs, besitzt jedoch entgegengesetzte Wertepräferenzen	erkennt die Eigenschaften und Werte des Webs und besitzt eben diese Wertepräferenzen

Tabelle 2.1: „Digital Visitors“ and „Digital Residents“ nach Kruse

	Gruppe 1: „Digital Visitors“	**Gruppe 2: „Digital Residents“**
Persönliches Gespräch vs. Kontakt über Internet	klar bevorzugt vs. eher problematisch	beides gleich attraktiv
Online-Petitionen als gelebte Demokratie	wird nicht akzeptiert	wird präferiert

Tabelle 2.2: Beispiele für Wertedifferenzen

mit ihm beschäftigen, derartige Wertedifferenzen beobachtet werden können und davon auszuegehen ist, dass viele Diskussionen dadurch auf Werte- und nicht auf Faktenebene stattfinden, auf der sich allerdings nur schwer konstruktiv und sachlich diskutieren lässt, kann dies eine Erklärung für die Schärfe der Diskussionen sein und dafür warum Diskussionspartner nicht zueinander können.

Für den Rahmen dieser Ausarbeitung ist eine andere Erkenntnis der Arbeit Kruses von Bedeutung: Die Gruppe der *Heavy User* des Webs, also derjenigen, die sich täglich intensiv mit dem Web auseinander setzen, es gut einschätzen und damit umgehen können, sollte nicht als homogene Gruppe angenommen werden, sondern es scheint deutliche Diffenrenzierungen u.a. eben auf der Werteebene zu geben. Da besonders bei den *Heavy Usern* des Webs davon ausgegangen werden kann, dass sie eine potentielle Zielgruppe der zu entwerfenden Architektur bilden, sollten die hier betrachteten Differenzen im Weiteren Entwicklungsprozess Beachtung finden.

2.2.2.2 Überblick

Als relevante Differenzierungsmerkmale potentieller Benutzer wurden beschrieben:

- die **Nutzungsrolle** mit den Ausprägungen **Produzent** vs. **Konsument**
- die **Nutzungsmotivation** mit den Ausprägungen **privat** vs. **beruflich**
- der **Nutzungskontext** mit den Ausprägungen **eher stationär** vs. **eher mobil**
- die **aktuelle Nutzung** mit den Ausprägungen **vorhanden** vs. **nicht vorhanden**
- das **technische Verständnis** mit den Ausprägungen **vorhanden** vs. **nicht vorhanden**
- die **Inhalte** mit zahlreichen Ausprägungen
- die **bevorzugte Öffentlichkeit** mit den Ausprägungen **primär öffentlich** vs. **primär teilöffentlich**
- das Verständnis **hinreichender Sicherheit** mit individuellen Ausprägungen
- bei Heavy Usern zusätzlich die **Wertepräferenzen** mit den Ausprägungen **Digital Visitors** vs. **Digital Residents**

Das Goal-directed Design gibt als Empfehlung an, dass mindestens vier der zu ihren Zielen befragten Benutzer jeweils eine Ausprägung verkörpern sollten, wobei eine Person auch mehrere Dimensionen bedienen darf (und in den meisten Fällen auch wird). Für die Befragung im Rahmen dieser Ausarbeitung wurden 12 Personen

ausgewählt, mit denen eine hinreichende Abdeckung aller Dimensionen erreicht wird. Die Verteilung der Ausprägungen ist:

- **Nutzungsrolle**: 7 × Produzent und Konsument, 1 × eher Produzent, 4 × eher Konsument
- **Nutzungsmotivation**: 7 × privat, 5 × beruflich
- **Nutzungskontext**: 4 × eher mobil, 8 × eher stationär
- **aktuelle Nutzung**: 3 × nicht vorhanden, 9 × vorhanden
- **technische Verständnis**: 7 × nicht vorhanden, 5 × vorhanden
- **bevorzugte Öffentlichkeit**: 3 × primär öffentlich, 9 × primär teilöffentlich
- **hinreichende Sicherheit**: 4 × geringe Anforderungen, 5 × genaue Anforderungen, 3 × genaue und hohe Anforderungen
- **Wertepräferenzen**: 5 × Digital Resident, 4 × Digital Visitor (3 Personen können nicht als *Heavy User* bezeichnet werden)

Die befragten Personen sind dabei zwischen 23 und 64 Jahre alt (Durchschnittsalter 32,6 Jahre), stammen beruflich sowohl aus Web-affinen Branchen (Web-Entwicklung, Web- und Kommunikationsdesign) als auch aus „durchschnittlich Web-zugänglichen“ Branchen (Literatur, Psychologie, Lehramt, Musik, Sozialpädagogik, Theologie, Philosophie, Betriebswirtschaftslehre) und besitzen allesamt einen unbeschränkten Zugang zum Web, weisen allerdings gänzlich verschiedene Verhaltensmuster, Sicht- und Herangehensweisen auf.

2.2.2.3 Methode

Als Methode für Benutzerpartizipation wird bewusst das persönliche Interview gewählt und *nicht* auf eine Datenerhebungsmethode zurückgegriffen, die durch das Web gestützt oder im Web durchgeführt wird (z.B. moderierter Diskurs in einem Weblog, Online-Fragebogen, ...). Zwar hätte man hierdurch die Reichweite im Sinne einer quantitativen Datenerhebungsmethode vervielfachen können, jedoch erscheint im Kontext der Ausarbeitung weniger die Quantität als die Qualität der Benutzerpartizipation relevant. Bei einer Beobachtung im Medium Web selbst wären die hinter den Online-Identitäten stehenden Subjekte (Benutzer) und deren Sinn- und Bedeutungsgebungen möglicherweise verborgen geblieben. Zudem hätte man durch die Nutzung des Webs bereits einen Teil potentieller Nutzer von der Partizipation ausgeschlossen. Nämlich diese Gruppe an potentiellen Nutzern, die aktuell keinen

Zugang zum Web haben oder das Web bewusst nicht nutzen, jedoch trotzdem Interesse an der Entwicklung jener zu entwerfenden Architektur haben mögen.

Das durchgeführte *qualitative Interview* nach Lamnek (1995) beschreibt als Oberbegriff eine Reihe von Interview-Methoden der qualitativen Sozialforschung, die sich in ihrem Grad der Strukturierung unterscheiden, jedoch allesamt eine Offenheit und weitgehende Nicht-Standardisierung der Befragungssituation gemein haben [Lam95]. Das Interview ist dabei weder in seinen Fragen noch in seinem Ablauf festgelegt, obgleich es sich natürlich um ein bestimmtes, zu erforschendes Thema dreht. Dies ermöglicht einen *«subjektnahen Einblick in Welterleben und Wirklichkeit der Befragten»* und dadurch auch neuartige oder überraschende Erkenntnisse hinsichtlich der Problemdomäne. Im Gegensatz zu einem *standardisierten Interview* wird eine Prädetermination der Inhalte durch den Durchführenden vermieden und der Befragte erhält die Möglichkeit an ihn heran getragene Konzepte selbst zu definieren. Lediglich ein Leitfaden mit zu berücksichtigenden Gesprächsinhalten dient während der Interviewsituation als Hilfe. Das Interview sollte inhaltlich umfassen:

- Gespräch über die persönliche Häufigkeit, den Kontext und die Intensität der Nutzung des Webs im Allgemeinen (dabei: wird das Webs als Bedrohung der Gesellschaft oder als Chance wahrgenommen?)
- Bei privatem Nutzungskontext: Gespräch über persönliche Erfahrungen mit Social Networks, mit E-Mail-Kommunikation oder anderen bevorzugten Formen des Teilens von Inhalten sowie deren Kommunikationspartner, konkrete Inhalte, Motivation, usw.
- Bei beruflichen Kontext: Gespräch über „geschäftstypisches“ Teilen von Inhalten im Web und dessen (vorgegebenen?) Rahmenbedingungen
- Immer: Gespräch über persönliche Probleme, Negativerfahrungen und deren Konsequenzen, ggfs. Alternativhandlungen außerhalb des Webs, Wünsche und Idealvorstellungen
- Gespräch über Kontrollverlust: Ist dieser während des Teilens bewusst oder wurde er sogar bereits erlebt?
- Gespräch über selbst-eingeschätzte Vertraulichkeit der eigen Inhalte und das Verständnis hinreichender Sicherheit

Diese nicht-standardisierte, sondern pro Gesprächspartner individuelle und qualitative Methode ist für die Nutzerpartizipation im Rahmen des hier durchgeführten Architekturprozesses besonders wertvoll, da vorab die zu erwartenden, unterschiedlichen Ansichten der Benutzer durch den Interviewer nur hätten erahnt werden

können und eine standardisierte Befragung die hier so relevante Sicht des Befragten ausgeblendet hätte.

Die Interviews wurden in ruhiger Büroumgebung in Einzelgesprächen ohne Zeitvorgabe durchgeführt. Sie dauerten zwischen 30 und 45 Minuten und sind in Auszügen im Anhang der Ausarbeitung dokumentiert.

2.2.2.4 Erkenntnisse

Der folgende Abschnitt fasst die Erkenntnisse der Benutzerbefragungen zusammen. Die Aussagen der Befragten wurden zu acht Themen-Clustern aufbereitet.

Über die Bereitschaft und Motivation des Teilens im Web

Im privaten Nutzungskontext schöpfen die befragten Personen ihre Motivation vor allem aus dem Bedürfnis nach sozialer Interaktion. Dem Web gegenüber aufgeschlossene Personen scheinen dabei die soziale Kommunikation über das Web gleichbedeutend zur Offline-Begegnung mit Freunden anzusehen. Dies geht einher mit der zuvor beschriebenen Beobachtung von Peter Kruse. Bersonders wichtig scheint neben dem reinen Mitteilen auch das Feedback zum geteilten Inhalt zu sein. Ein „Danke!“ oder „Das gefällt mir!“ als Reaktion auf einen geteilten Inhalte erfreut, verleiht Anerkennung unter Freunden und motiviert soziale Beziehungen auf diese Weise zu pflegen. Feedback auf einen geteilten Inhalt kann allerdings auch indirekt erfolgen: Personen, die im Interview angaben Artikel für die Wikipedia zu verfassen, schöpfen Motivation aus der Lebhaftigkeit der anschließenden Diskussionen rund um den Artikel oder aus seinen Abrufstatistiken. Auch ermöglicht das Teilen im Web neue Formen der sozialen Interaktion überhaupt erst, was als weitere Motivation gesehen kann: Zum Beispiel ist das Teilen von Inhalten mit mehreren Personen gleichzeitig im Web besonders einfach (beispielsweise das Teilen der Fotos vom Wochenendausflug im Freundeskreis) und wäre in dieser Form im Offline-Leben erheblich langwieriger, wenn überhaupt praktikabel: Hier müsste ein Termin gefunden werden, bei dem gemeinsam auf die Fotos geschaut würde. Diskussionen könnten nur an diesem Termin stattfinden und das *gemeinsame* Teilen wäre nach diesem Zeitfenster vorbei. Im Web können Dialoge dagegen länger leben; auch wenn sie zeitweise „pausieren“, können Gespräche jederzeit wieder aufgenommen werden — eben dann, wenn einer der Gesprächspartner es gerade für sinnvoll hält.

Im beruflichen Kontext erzwingt die Ausübung des Berufes in vielen Fällen die Nutzung des Webs zur Kommunikation. Das Teilen von Inhalten (in diesem Kontext

häufig: Dokumente) im Web ist dabei *ein* Kommunikationsmittel, welches die gemeinsame Arbeit mit ihnen vereinfacht. Dokumente müssen von anderen durchgesehen und annotiert werden oder dienen als Grundlage, auf der Arbeit anderer Personen aufbauen kann. Das Teilen impliziert daher eher einen Handlungsauftrag als im privaten Kontext und bewegt Inhalte an den Anfang einer gemeinsam durchgeführten Prozesskette. Das Feedback zu geteilten Inhalten spielt sich dann allerdings auch eher auf professioneller Ebene ab und ist inhaltlich begründet. Das Teilen selbst wird als selbstverständlich für eine Zusammenarbeit angesehen.

Über den Ablauf des Teilens

Aus den Schilderungen der befragten Personen lässt sich ein vollständiger Ablauf des Teilens im Web in verschiedene Phasen gliedern:

- Zu Beginn steht die **Teilungsabsicht**. Es entsteht die Idee, das Bedürfnis oder der Zwang eine Information mit anderen zu teilen. Diese Information kann zu diesem Zeitpunkt schon vorliegen (z.B. gerade aufgenommenes Foto) oder nicht (z.B. Text, der noch geschrieben werden muss).
- Im Rahmen einer **Auswahl des Kommunikationsmittels** wird eine konkrete Anwendungen des Webs ausgewählt, die für das Teilen genutzt werden soll. Rahmenbedingung bei dieser Entscheidung kann auch der gegenwärtige Zugang zum Web sein. Ist man unterwegs, hat man auf einem mobilen Gerät mitunter nur Client-Software für bestimmte Web-Anwendungen installiert.
- Als Vorbereitung auf das eigentlich Verfassen des Inhaltes kann eine **Web-Recherche** eingeschoben werden, in der ggfs. Links zusammengetragen werden, Bilder, Karten, Adressen und alle anderen Informationen, die man z.B. zum Verfassen einer Nachricht benötigen könnte.
- In der **Entwurfsphase** entsteht nach und nach der zu teilende Inhalt. Bilder erhalten ggfs. Bildunterschriften oder werden mit Metadaten (z.B. mit Geodaten) versehen, Texte werden geschrieben. Während der Inhalt verfasst wird, bleibt er jedoch in der Regel noch unveröffentlicht und kann nur vom Autor betrachtet und verändert werden.
- Wird ein Entwurf als fertig erachtet, ist je nach Anwendung eine **Auswahl der Empfänger** notwendig, um ihn mit diesen zu teilen. Bei vollständig öffentlichen Web-Anwendungen wie Twitter entfällt die Auswahl, da hier alle Inhalte der Web-Öffentlichkeit zugänglich sind[7] und keine Einschränkung pro

[7] bei privaten Accounts ist diese Öffentlichkeit eine Teilöffentlichkeit mit autorisierten Teilhabern

Inhalt vorgenommen werden kann. In Social Networks können als Empfänger mitunter auch Gruppen dienen, die verschiedene Kontakte zusammenfassen.

- Beim **Veröffentlichen** wird der Inhaltsentwurf schließlich an die Web-Anwendung übertragen und ist nun dort für autorisierte Personen einsehbar; ggfs. werden die Empfänger über die Veröffentlichung direkt benachrichtigt (z.B. per E-Mail, SMS, RSS oder Push Notification Service).
- Während der *Lebensphase* des Inhaltes haben autorisierte Empfänger nun häufig die Möglichkeit den Inhalt zu kommentieren, Feedback dazu abzugeben und damit in einen Dialog zu treten, der wiederum viele geteilte Inhalte beinhalten kann.

Auf die letzte Phase, die als *Lebensphase* eines Inhaltes, bezeichnet wurde und die Implikationen des Existierens von Inhalten im Web referenziert wird im Abschnitt „Über die Inhalte des Teilens“ näher eingegangen.

Über die Teilhaber

Das Teilen von Inhalten ist eine gerichtete Kommunikation. Es existieren (bevorzugte) Empfänger, mit denen ein eigener Inhalt initial geteilt werden soll und die vom Urheber bestimmt werden.

Im beruflichen Kontext sind die Empfänger eines Inhaltes meist bekannt, da bereits Geschäftsbeziehungen zu ihnen existieren. Die Identität der Kommunikationspartner wird teilweise durch besonders Vorkehrungen (z.B. das digitale Signieren von Inhalten) gesichert und es besteht in der Regel persönlicher Kontakt außerhalb der Anwendung, die zum Teilen genutzt wird.

Im privaten Kontext kann dies anders aussehen. Zwar machen Familie, Freunde und Bekannte einen Großteil der privaten sozialen Online-Kontakte aus, mit denen eigenen Inhalte geteilt werden, jedoch entstehen gerade in Social Networks auch neue Kontakte zu unbekannten Personen, deren Identität nicht gesichert werden kann. Das zeitlose Cartoon des *The New Yorker*-Magazins aus dem Jahr 1993 bringt das immer noch im Web existierende Identitätsproblem auf den Punkt: „On the Internet, nobody knows you’re a dog.“

Doch hemmt viele Nutzer der Fakt nicht, dass die Identität des Gegenüber nicht gesichert sein kann. Das Teilen mit unbekannten Personen findet statt, wenn auch in veränderter Form. So bilden einige der befragten Personen Gruppen aus ihren Online-Kontakten und erlauben diesen spezielle, beschränkte Sichten auf eigene Inhalte. Gute Freunde und die Mitglieder der Familie dürfen meist mehr Inhalte einsehen und

"On the Internet, nobody knows you're a dog."

Abbildung 2.1: Identitäten verbergen sich im Web oftmals hinter Pseudonymen und sind selten gesichert.

kommentieren, als entfernt Bekannte oder gänzlich unbekannte Personen. Weitere Erkenntnisse „über die Privatheit des Teilens“ werden in einem folgenden Abschnitt näher dargestellt.

Die Gruppe der Teilhaber kann weiterhin in zwei Gruppen unterteilt werden: Die **primären Teilhaber**, die vom Urheber eines Inhaltes als Empfänger bestimmt wurden, und die unbekannte Gruppe **sekundärer Teilhaber**, die in Besitz oder Kenntnis eines Inhaltes durch die Weitergabe durch einen der primären Teilhaber gelangt sind. Denn das Web lädt dazu ein, empfangene Inhalte an weitere Personen weiterzuleiten. Sind diese nicht durch eine entsprechende Maßnahme geschützt (z.B. durch das Vorhandensein eines Digital-Rights-Managements, welches die Nutzung nur auf autorisierter Hardware ermöglicht), kann man festhalten: Ein Urheber eines eigenen Inhaltes kann pro Inhalt bevorzugte, primäre Teilhaber auswählen und muss jedoch davon ausgehen, dass sich sein Inhalt über diese Gruppe hinaus im Web verbreitet. Diese Tatsache verleitet einige der befragten Personen dazu, auch in augenscheinlich geschlossenen Gruppen keine Inhalte zu teilen, die nicht für eine erweiterte Öffentlichkeit bestimmt sind.

2.2.2.5 Über die Orte des Teilens

Geteilt werden Inhalte in der Regel innerhalb von Web-Anwendungen, doch lässt sich diese Annahme nach den Gesprächen näher differenzieren: Im privaten Kontext spielen sich in der Gruppe der Befragten die Online-Aktivitäten überwiegend in größeren Social Networks ab: Aus der Nutzung mit Twitter wurde beispielweise 6 Mal berichtet, davon 2 Mal mit nicht-öffentlichen Accounts. Aktive Facebook-Accounts besitzen 9 der 12 Personen. Während die Kommunikation und das Teilen von Inhalten via Twitter dabei meist als „ungerichtet" beschrieben wurde, also ohne genauen Empfänger vor Augen, und die Anzahl potentieller Teilhaber (Anzahl der *Follower*) sich teilweise auch in Bereichen um 1000+ bewegte, ist der Empfängerkreis geteilter Inhalte bei Facebook im Durchschnitt vergleichsweise gering und weist auch wesentlich stärkere sozialen Beziehungen auf. Hier existieren — im Gegensatz zum teilweise anonym wirkenden Twitter — tatsächlich Freund- und Bekanntschaften.

Services wie Facebook oder Twitter ermöglichen es ihren Nutzern auch, ihre dort erzeugten und gepflegten Online-Identitäten in vielen weiteren Web-Kontexten zu nutzen. Nach dem *Single Sign On*-Prinzip[8] können diese Accounts auch als Login für weitere Web-Anwendungen verwendet werden. Die eigenen Kontakte aus Facebook oder Twitter, der so genannte *Social Graph*, sind dann auch in diesem neuen Kontext als Information nutzbar. Zwar stellen die meisten Social Networks in Bezug auf die Online-Identität ihrer Benutzer derzeit immer noch Insellösungen dar und existieren separat nebeneinander, doch können durch derartige Verbindungen zwischen Web-Anwendung ähnliche Teilungskontexte in ansonsten getrennten Social Networks entstehen. In den Gesprächen wurde hier z.B. auf den Photo-Sharing-Service *Instagram*[9] eingegangen, der einerseits den Import von Facebook-Kontakten in das eigene Netzwerk erlaubt, andererseits aber auch auf Wunsch des Benutzers innerhalb von Instagram geteilte Fotos direkt im Facebook-Profil des Benutzers veröffentlicht (Abbildung 2.2). Auf diese Weise entstehen geteilte Inhalte an mehreren Stellen gleichzeitig im Web und sind für einen ähnlichen Empfängerkreis auch an mehreren Stellen im Web gleichzeitig einsehbar und kommentierbar. Es stellt sich die Frage: Welcher Inhalt ist dann der Referenzinhalt und welcher eine Kopie davon?

Neben den bekannten, großen Social Networks wurden darüber hinaus zahlreiche kleinere, eher themenvertikale Web-Anwendungen als Orte des privaten Teilens im Web genannt: Web-Anwendungen zu wissenschaftlich/fachlichen Themen (Astronomie, Psychologie, Philosophie, ...) oder zu Hobbys und privaten Interessen (Wein, Kochen, Segeln, Garten, ...). Diese werden meist über einen Browser genutzt, verlangen

[8] bei Twitter *Twitter @anywhere* und bei Facebook *Facebook Connect* genannt
[9] http://instagr.am/

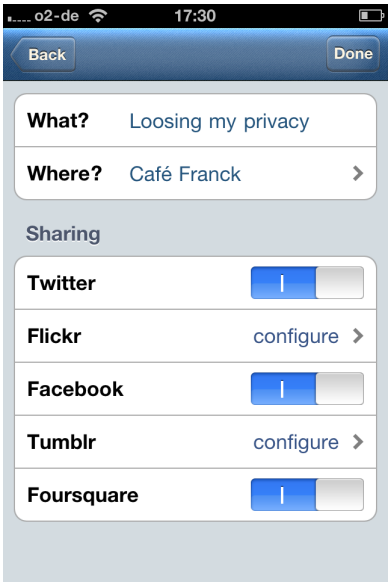

Abbildung 2.2: Die iOS-Software Intagram veröffentlicht aufgenommene Fotos direkt bei verschieden Social Networks

teilweise aber auch spezielle Client-Software, die dann oft betriebssystemübergreifend angeboten wird: Der Musik-Dienst *Spotify* ist mit seinem angeschlossenen Social Network beispielsweise nur über eine spezielle Player-Software zu bedienen[10]. Ähnlich spielt sich auch die Interaktion mit dem von Apple betriebenen Social Network *Ping!* vollständig in der Apple Software *iTunes* ab[11].

Aus dem beruflichen Kontext wurden sowohl die zu erwartenden, klassischen *Business Social Networks* als bevorzugte Orte des Teilens genannt (XING[12], LinkedIn[13], ...), aber auch File-Sharing-Dienste wie Dropbox[14] oder CloudApp[15], die durch ihre Client-Software eine tiefe Betriebssystemintegration besitzen und sich somit einfach und durch native Betriebssysteminteraktion in bestehende Arbeits-Workflows integrieren lassen. Hier geschieht das Teilen fast unbewusst nebenbei: Legt man bei der Nutzung von Dropbox eine Datei in einen bestimmten Ordner auf der eigenen Festplatte, wird diese automatisch durch eine im Hintergrund laufende Anwendung ins Web hochgeladen und von dort auf die lokalen Rechner autorisierter Teilhaber kopiert. Es existiert hier quasi ein autorisiertes „Abonnement“ einer Informationsquelle (Ordner), welches das Teilen von Inhalten stark vereinfacht, weitgehend automatisiert, aber vor allem es völlig aus der sonst für das Web typischen Interaktion mit

[10] http://www.spotify.com/int/
[11] http://www.apple.com/de/itunes/ping/
[12] https://www.xing.com/
[13] http://www.linkedin.com/
[14] https://www.dropbox.com/
[15] http://www.getcloudapp.com/

einer Web-Anwendung in einem Browser herauslöst.

Bemerkenswert in Hinblick auf die im ersten Teil dieser Ausarbeitung angestellte Überlegung zur Lösung des Zentralismusproblems fiel ferner auf: In beiden Kontexten — privat und beruflich — werden dezentrale Social Networks (diese werden in Abschnitt 2.2.3 näher behandelt) derzeit kaum genutzt. Selbst Gesprächspartner, die hohe Anforderungen an die Vertraulichkeit ihrer Web-Kommunikation besitzen, äußerten Zweifel an der Praktikabilität jener Peer-2-Peer Social Networks. Die technische Hürde eigene Infrastruktur in Form eines Servers (Hard- und Software) zu betreiben, um an einem Social Network teilnehmen zu können, wäre nichts was der Alltagsnutzer des Webs ohne Anstrengung bezwingen könne. Diesen Punkt gilt es daher im weiteren Verlauf des Architekturprozesses kritisch zu diskutieren.

Über die Inhalte des Teilens

Die in den Interviews beschriebenen geteilten Inhalte entsprachen dem allgemeinen Bild, das aus Social Networks bekannt ist: Die befragten Personen teilen vorwiegend Texte unterschiedlicher Länge (kurze Statusnachrichten vs. längere Weblog-Einträge), diese können zudem Links enthalten, jedoch eher selten Formatierungen (Rich Text). Fotos werden oft auch direkt als Album geteilt, welches dann aus mehreren Fotos besteht. Sie werden dazu vom lokalen Rechner oder mobilen Endgerät hochgeladen. Auch im Web gefundene Fotos und Grafiken werden geteilt, deren Nutzungsrechte allerdings teilweise zweifelhaft oder unbekannt sind. Geteilte Videos sind entweder mit entsprechender Hardware selbst aufgenommen, besitzen dann immer häufiger bereits HD-Qualität und werden meist direkt mit Funktionen des Aufnahmegeräts im Web veröffentlicht, oder sie befinden sich bereits im Web. Dann wird allerdings meist nur noch der Link geteilt, nicht das Videomaterial selbst. Dieses bleibt an zentraler Stelle und wird von dort in verschiedene Kontexte eingebettet.

Inhalte können jedoch offensichtlich über ihre Datentypen hinweg verschiedene Eigenschaften besitzen: Sie können „final“ sein und sind dann nur noch zum Ansehen bestimmt oder sie können für eine (kollaborative) Bearbeitung bestimmt sein. Sie können zeitkritisch sein und sollten dann den Empfänger auch in einer garantierten Zeit erreichen. Ihre tatsächliche, erfolgreiche Zustellung an den Empfänger kann von Bedeutung sein oder es kann mitunter auch egal sein, ob ein Inhalt den ausgewählten Empfänger überhaupt erreicht oder von ihm gelesen wird. Inhalte können nur in einem endlichen Zeitraum für seine Empfänger von Interesse sein und verlieren danach vollständig an Informationswert, sind vergänglich. Sie können auch eine vorbestimmte Lebensdauer besitzen nach der sie nicht mehr existieren sollten. Die Möglichkeit einer *Lebenszeit* eines Inhaltes böten allerdings nur sehr wenige aktuelle

Web-Anwendungen an.

Über die Privatheit des Teilens

Während der Interviews wurden verschiedene Aussagen über die „angenommene Vertraulichkeit“ eigener Inhalte getätigt. Der Schutz der eigenen Inhalte und der „digitalen Privatsphäre“ wird unter den Befragten derzeit verschieden praktiziert.

Es sind Unterschiede erneut vor allem in der Differenzierung beruflich vs. privat sichtbar: Im beruflichen Kontext wird häufiger angenommen, dass kommunizierte Inhalte vertraulich sind, da sie der Erreichung eines Geschäftsziels dienen und ihre Veröffentlichung diesem entgegenstehen könnte. Diese Wahrnehmung spiegelt sich jedoch nur selten auch in den angewandten Maßnahmen zum Schutze der Inhalte wider: Nur wenige Personen verschlüsseln beispielsweise E-Mails mit vertraulichem Inhalt oder sichern komprimierte Archive, die sie im Web hochladen, mit Passwörtern. Dies sei zu kompliziert, würde zu Kompatibilitätsproblemen auf verschiedenen Systemen führen (genannt bei der Verschlüsselung von E-Mails per GPG) oder schlicht „nicht in den Workflow passen“ oder den Spaß bei der Arbeit schmälern. Es spielt offensichtlich auch die Erfahrung der Personen eine Rolle: Wer bisher noch keinen Missbrauch eigener Inhalte erlebt hat, geht aus dieser Beobachtung sorgloser mit ihnen um. Lediglich auf HTTPS-Verbindungen wird von erfahrenden Benutzern geachtet, wenn vertrauliche Dateien zwischen dem lokalen Rechner und einer Web-Anwendung ausgetauscht werden, z.B. beim Online-Banking oder -Shopping.

Im privaten Kontext gehen die Annahmen über die Vertraulichkeit der Inhalte und die Privatheit der Kommunikation als solcher auseinander: Auf der einen Seite gibt es die Gruppe an Personen, die sehr gerne öffentlich Inhalte teilt und nicht nur ihre Privatsphäre, sondern teilweise auch ihre Intimspähre dem Web öffnet. Das Wissen darüber, dass ein bestimmter Inhalt von der Gesellschaft wohl eher als privat angesehen wird, hindert sie nicht daran diesen öffentlich zu machen. Vor allem bei offenen Social Networks wie Twitter ist dieses Phänomen des freizügigen Umgangs mit der eigenen Privatsphäre zu beobachten. Daneben existiert eine zweite Gruppe an weniger öffentlich lebenden Personen, die als privat erkannte Inhalte auch tatsächlich nur in einer kleinen Öffentlichkeit teilt. Dass dies dann aber auch technisch abgesichert und „wasserdicht“ erfolgt, scheint weniger wichtig. Sie vertrauen den meisten Web-Anwendungen, dass ihre Daten nicht ohne Zustimmung öffentlicht gemacht werden, sondern nur dem bestimmten Freundeskreis zugänglich sind. Erkenntnisse über den Ursprung eines solchen Vertrauens beschreibt der folgende Absatz „Über die Vertrauenswürdigkeit einer Architektur“. Schließlich treten die Anhänger einer dritten Gruppe dem Web mit kritischer Skepsis gegenüber und sehen an vielen

Stellen Bedrohungen ihrer Identität, ihrer Privatsphäre und der eigenen Inhalte. Ein grundsätzliches Misstrauen gegenüber allen im Web angebotenen Anwendungen begleitet die Interaktion mit diesen und prägt den Umgang mit eigenen Inhalten. Personen dieser Gruppe verkehren im Web gerne unter Pseudonym, blockieren Tracking-Möglichkeiten ihrer Online-Aktivität oder verrauschen oder verwischen ihre Spuren im Web durch „Privatsphäre-sichernde Software“. Das öffentliche Auftreten unter dem eigenen Namen liegt ihnen fern, zum Teilen eigener Inhalte (z.B. im Freundeskreis) werden besondere Vorkehrungen (Verschlüsselung von Daten und Verbindung, Benutzung von VPN-Netzen, Benutzung des Tor-Netzwerkes[16], ...) getroffen.

Die Einschätzungen des Wertes der eigenen digitalen Privatsphäre gehen in der Gruppe der befragten Personen deutlich auseinander: Die Ansprüche an technische Maßnahmen zum Schutze der eigenen Inhalte reichen von „eigentlich egal“ bishin zu „zwingend erforderlich“ mit konkreten Anforderungen. Interessant ist jedoch eine Gemeinsamkeit unter all diesen Teilgruppen: Das Vorhandensein und die Nutzung von Personae. Damit ist gemeint, dass Personen in unterschiedlichen Kontexten unter verschiedenen Identitäten auftreten, dabei z.B. verschiedene Namen benutzen, verschiedene soziale Kontakte pflegen und auch Persona-spezifische Interaktionsmuster besitzen. So kann z.B. eine Person mit seinem Klarnamen in wohldefinierter Teilöffentlichkeit nur mit seinen Freunden kommunizieren, gleichzeitig aber auch eigene Inhalte mit der Web-Öffentlichkeit — dann ggfs. unter Pseudonym — teilen. Betrachter können dabei mitunter Zusammenhänge zwischen Personae erkennen und beide mit einer real existierenden Person in Verbindung bringen, die Personae können jedoch auch so disjunkt sein, dass sich eine Beziehung und die dahinter stehende Person nicht erschließen lassen. Die Nutzung mehrerer Identitäten, von denen meist letztendlich von außen betrachtet keine als gesichtert angenommen werden kann, ist eine schon lange populäre Spielart des Webs und gehört fest zur Kultur des Internets. Es überrascht daher nicht, dass gerade beim Teilen eigener Inhalte eine starke Nutzung von Personae zu beobachten ist. Dies ist jedoch eine wichtige Erkenntnis für den Entwurf einer entsprechenden Architektur.

Über die Negativerfahrungen beim Teilen

Im Zusammenhang mit der Einschätzung der Privatheit eigener Inhalte und den Berichten mit deren Umgang wurden oft Negativerfahrungen beim Teilen von Inhalten im Web angesprochen. In den meisten Fällen ging es hier darum, dass unautorisierte Personen Zugriff auf eigene Inhalte erhalten haben und dies sich negativ auf einen

[16] http://www.torproject.org/

geschäftlichen Erfolg ausgewirkt oder soziale Beziehungen strapaziert hat. Kaum verhinderbar erscheint den Befragten aktuell die Weitergabe von Inhalten im Web. Zwar bringe man seinen sozialen Kontakten das nötige Vertrauen entgegen, dass Inhalte, die als solche gekennzeichnet oder benannt sind, nicht ohne Rückfrage an Dritte weitergereicht werden, doch schützt einem kein Softwaresystem davor — ganz im Gegensatz: Anwendungen wie E-Mail machen es denkbar einfach empfangene Inhalte einfach an neue Empfänger „weiterzuleiten“. Diese unautorisierte Weitergabe von Inhalten steuern zu können, war besonders ein Wunsch der Personen, die sich kritisch mit dem Web auseinandersetzen oder es aus beruflichen Gründen mit mehr Vorsicht nutzen.

Darüber hinaus wurden Negativerfahrungen als Konsequenz von Versehen berichtet: Nutzt eine Person beispielsweise mehrere Benutzerkonten bei einer Web-Anwendung (z.B. Twitter), so sei es „schon oft passiert“, dass Inhalte versehentlich unter einem anderen Account veröffentlicht wurden und dadurch entweder die falschen Empfänger erreicht oder dem dann tatsächlichen Absender „falsche Worte in den Mund gelegt“ wurden. Durch derartige Verwechselungen können auch Beziehungen zwischen Personae aufgedeckt werden! Bei Web-Anwendungen, die eine Empfängerauswahl für das Teilen voraussetzen, ist ein weiteres häufiges Versehen die Auswahl nicht beabsichtigter Empfänger (mit ähnlichen Konsequenzen). Weiterhin beschrieben die befragten Personen von Negativerfahrungen mit versehentlich verwechselten Inhalten. Gerade bei ähnlich benannten Fotos oder Dokumenten, die per Dateiupload vom lokalen Rechner ins Web überträgt, sei eine Verwechselungsgefahr gegeben. Die Folgen können je nach Verwechselung „belanglos“, „glimpflich“, „peinlich“ sein, aber auch Freundschaften oder geschäftliche Beziehungen bedrohen.

Besonders aus offenen Social Networks wie Twitter konnten einige Befragte zudem von Erfahrungen mit Identitätsdiebstahl bzw. -betrug berichten. Nicht selten existieren dort (zeitweise) Benutzer-Accounts unter Namen prominenter Personen, die vorgeben offizieller Twitter-Account jener Person zu sein. Dort durch Dritte veröffentlichte Inhalte können das Ansehen der Person in der Öffentlichkeit gefährden, wenn die kriminelle Handlung nicht für jeden direkt offensichtlich ist. Eine ähnliche Spielart des Identitätsmissbrauchs bezeichnet man im Web-Kontext auch als *Nicknapping*. Das Auftreten einer Person unter Pseudonym, welches in diesem Kontext jedoch bereits durch eine andere Person regelmäßig verwendet wird, ist ein Problem in vielen Kommunikationskontexten wie dem IRC, in Foren, Mailinglisten oder im Usenet.

Über die Vertrauenswürdigkeit einer Architektur

Ein zentrales Ziel der Befragungen war es, Ansichten und Erfahrungen hinsichtlich der Vertrauenswürdigkeit von Web-Architekturen und -Anwendungen zu erkennen. Denn dies soll eine zentrale Eigenschaft der neu zu entwerfenden Architektur sein: Sie soll aus sich heraus vertrauens*würdig* sein, damit eine Nutzung tatsächlich eintritt. Anhand welcher Eigenschaften werden nun also Web-Anwendungen als vertrauenswürdig anerkannt und lassen sich diese auf deren technische Basis, die Architektur, übertragen?

Vertrauen in Web-Anwendungen dies sei im privaten Kontext vor allem dann notwendig, wenn es im Web um Finanzgeschäfte geht (viele Personen sprachen hier das Online-Banking an) oder um das Einkaufen im Web. Denn in diesen Situationen ist entweder reales Geld im Spiel und/oder man interagiert mit unbekannten Kommunikations- und Geschäftspartnern. Doch auch unter bekannten Kommunikationspartnern kann Vertrauen in das Kommunikationsmittel notwendig sein: Denn immer dann, wenn man im Web so genannte „sensible Daten“ (Bankverbindung, Kreditkartennummern, Adressen, ...) austauscht, möchte man diese gerne in sicheren Händen sehen und vor Missbrauch geschützt wissen. Da die endgültige Gewissheit, dass ein Missbrauch ausgeschlossen ist, im Web nicht erlangt werden kann, wird hier *Vertrauen* notwendig. Man erwartet einen positiven Verlauf bei gleichzeitiger Ungewissheit. Die Frage „Was hilft dabei Vertrauen aufzubauen?“ beantwortete die befragte Gruppe auf drei verschiedene Arten, die für den weiteren Architekturprozess eine wertvolle Erkenntnisgrundlage bilden:

1. **Vertrauenswürdigkeit entsteht durch Transparenz und Nachvollziehbarkeit.** Sofern Web-Anwendungen ihre Architektur und interne Arbeitsweise offenlegen und nachvollziehbar[17] dokumentieren, können Benutzer mit entsprechendem Fachwissen die Sicherheit der eigenen Inhalte innerhalb dieser Anwendung beurteilen. Dieses Urteil bleibt subjektiv und unbestätigt, da eine endgültige Prüfung der tatsächlichen Arbeitsweise meist nicht möglich ist. Dennoch ist dies für technisch versierte Benutzer eine erste Grundlage, auf der eine Vertrauensbeziehung durch einen „Vertrauensvorschuss“ aufgebaut werden kann. Leider offenbaren wenige Web-Anwendungen eine derartige Transparenz, die eine zufriedenstellende Beurteilung zulässt, und nur ein Bruchteil der Benutzer kann zudem mit jenen technischen Informationen überhaupt etwas anfangen. Aus diesem Grund gilt für den Großteil der befragten Personen:

2. **Vertrauenswürdigkeit entsteht durch das Urteil externer Autoritäten.** Ist man nicht selbst dazu in der Lage ein eigenes Urteil hinsichtlich der Vertrau-

[17] siehe hierzu auch Abschnitt 2.2.1.3

enswürdigkeit aufgrund einer technischen Bewertung einer Anwendung zu fällen, entweder weil die Anwendung ihre technische Grundlage nicht nachvollziehbar kommuniziert oder weil man kein entsprechendes Fachwissen besitzt, dann kann als Grundlage der eigenen Vertrauensbeziehung das Urteil externer Autoritäten übernommen werden. Damit sind alle Individuen aus Freunden, Familienmitgliedern, Redaktionen von Fachmagazinen und -zeitschriften, TV-Moderatoren und viele weitere Personen des öffentlichen und nicht-öffentlichen Lebens gemeint, deren Urteil man in diesem speziellen Fall eher vertraut als seinem eigenen. Dies kann durch das Fachwissen der Person begründet sein, aber auch durch eine gute Freundschaft zu ihr, durch die Bekanntheit einer Person in der Öffentlichkeit oder durch das Vertrauen, dass man ihr in anderen Situationen bereits entgegenbringt. In persönlichen Gesprächen, Zeitschriften, TV-Beiträgen, Weblogs oder Artikeln im Web stößt man auf diese Urteile, kann mitunter die Bewertungsgrundlage hinterfragen, diese für sich selbst beurteilen und dann der Quelle ggfs. Vertrauen schenken. Es entsteht erneut ein Vertrauensvorschuss, der zur Nutzung einer unbekannten Anwendung führen kann. Inwiefern eine Person dann feststellen kann, ob dieser Vorschuss begründet war, dies zeigt die dritte Alternative, um die Vertrauenswürdigkeit einer Web-Anwendung zu beurteilen:

3. **Vertrauenswürdigkeit entsteht durch Erfahrung.** Die Aussage des bereits in Abschnitt 2.2.1.2 angeführten Zitats von Niklas Luhmann *«Vertrauenswürdig ist, wer bei dem bleibt, was er bewusst oder unbewusst über sich selbst sichtbar gemacht hat.»* spiegelt sicht deutlich auch in den Interviews mit potentiellen Nutzern wider: Fehlen sowohl die Grundlage, um auf Basis von Transparenz und Nachvollziehbarkeit eine Vertrauensbeziehung aufzubauen, als auch Urteile externer Autoritäten, kann als dritter Weg zur Beurteilung ihrer Vertrauenswürdigkeit nur noch die eigene Erfahrung mit einer Anwendung dienen. Einige der Befragten beschrieben wie sie sich der Nutzung gänzlich unbekannter Web-Anwendungen schrittweise nähern: Sie registrieren sich erst unter einer Scheinidentität und interagieren mit dieser „gefahrlos“ innerhalb der Anwendung. Sollten die Daten veruntreut oder missbraucht werden, ist dies ohne Konsequenz. Wenn sie dabei dann den Eindruck gewinnen, dass eine Anwendung die Erreichung eines eigenen Ziels unterstützten könnte, so probieren sie Schritt für Schritt die angebotenen Funktionen der Anwendung aus und überprüfen immer so gut es geht das Verhalten der Anwendung. Wenn sie dann feststellen, dass die vorher kommunizierte Verhaltensweise mit der erlebten über einen gewissen Zeitraum der Nutzung übereinstimmt, so kann davon ausgegangen werden, dass dies wohl auch für die künftige Nutzung der Fall sein wird. Diese positive Erwartung des Ereignisverlaufs ist entstandenes Vertrauen. Auf die

gleiche Art und Weise kann die Rechtfertigung eines Vertrauensvorschusses geprüft werden, der auf Basis der ersten beiden Herangehensweisen erfolgt ist. Die eigene, konstant positive Erfahrung mit einer Anwendung gewinnt mit der Zeit der Nutzung stetig an Bedeutung für die Vertrauensbeziehung zur Anwendung. Während die ersten beiden Herangehensweisen sehr hilfreich bei der initialen Beurteilung einer Anwendung sein können, erhält die eigene Erfahrung eine langfristige Vertrauensbeziehung und Vertrauenswürdigkeit aufrecht.

Eine Vertrauensbeziehung zu einer Web-Anwendung kann allerdings auch erschüttert werden, deren Vertrauenswürdigkeit leidet dann je nach Ursache unterschiedlich stark. In diesem Kontext blieben einigen Personen die brisanten Datenschutzprobleme des deutschen Social Networks *StudiVZ* im Herbst 2006 in Erinnerung [ho06]: Durch geringe Änderungen einer URL konnten explizit als nicht-öffentlich markierte Inhalte der Mitglieder von unautorisierten Personen abgerufen werden. Diese Sicherheitslücke wurde durch Weblogs verbreitet und durch StudiVZ zügig behoben, stellte allerdings erst den Anfang der Aufdeckung weiterer Datenlecks dar, die über Wochen in den Medien dokumentiert und kritisch diskutiert wurden. Diese über einen relativ langen Zeitraum derart negativen Bewertungen der Datensicherheit von vielen unterschiedlichen Quellen hat die Mitgliederanzahl des Social Networks einbrechen lassen. Bis heute, gute vier Jahre später, ist für einige der befragten Personen eine Nutzung des StudiVZs immer noch ausgeschlossen, auch wenn die Datensicherheit der Anwendung mittlerweile durch den TÜV-Süd bestätigt wird [Stu10]. Hier wiegen die eigenen negativen Erfahrungen offenbar stärker als eine externe Autorität, der in anderen Kontexten (Verkehrssicherheit von Fahrzeugen) allgemein vertraut wird.

Der Umgang mit dem Beispiel „StudiVZ“ legt auch die Existenz eines weiteren Faktors nahe, der die Vertrauenswürdigkeit einer Web-Anwendung beeinflussen kann: Das dahinter stehende Team. Dies sei laut den befragten Personen bei StudiVZ seitdem nicht vollständig ausgewechselt und daher könne die Anwendung nun zwar geprüft sein, wer jedoch einmal solche „haarsträubenden Fehler“ begeht, dem ist dies noch einmal zuzutrauen.

Über das technische Verständnis

Immer wieder ist man als Benutzer des World Wide Webs mit seiner zugrunde liegenden Technik konfrontiert. Zwar setzt die Nutzung lediglich die korrekte Bedienung eines Computers sowie einer Browser-Software voraus, doch kann letztere mitunter bereits komplex werden: In den Interviews konnten nur knapp die Hälfte der

Personen benennen wofür der Präfix `http://` steht, der jeder WWW-Adresse in der Adresszeile des Browser vorangestellt wird. HTTPS, Zertifikate, Cookies, GPG-Verschlüsselung, Public Keys und Prüfsummen — die Technik-Landschaften des Webs und gerade die Techniken, die eingesetzt werden, um *sicher* über das Web zu kommunizieren, sind umfangreich und selbst für Fachleute eine Herausforderung in der Einrichtung und korrekten Nutzung.

Diese Vermutung offenbarte sich auch in den Gesprächen mit potentiellen Benutzern. Nur die Personen, die entsprechendes Fachwissen besitzen und sich professionell mit dem Web auseinander setzen, konnten fundiert über Sicherheitsfragestellungen im Web diskutieren. Im Gespräch mit „normalen“ Anwendern hingegen verblieb das Gespräch mit seinen Diskussionselementen meist auf Anwendungs- und Aufgabenebene. Das nicht vorhandene tiefe technische Verständnis in einem bedeutenden Teil der potentiellen Nutzergruppe stellt zu einem späteren Zeitpunkt gerade in Hinblick auf die Gestaltung der Nachvollziehbarkeit der Arbeitsweise der Architektur eine Herausforderung dar.

2.2.2.6 Über Wünsche und Vorstellungen einer alternativen Architektur

In offenen Gesprächen über die persönlichen Vorstellungen einer alternativen Architektur zum Teilen eigener Inhalte im Web, ergaben sich zahlreiche Impulse und Wünsche. Davon besonders herauszustellen sind:

«*So „sicher“ eine alternative Architektur auch sei, sie solle nicht wesentlich komplizierter zu bedienen sein als aktuell genutzte Anwendungen (v.a. Social Networks).*» Dies steht womöglich einerseits mit der nicht immer gegebenen Technik-Affinität in Verbindung, mag aber auch die Anforderung an einen *Joy of Use* darstellen: Viele Benutzer scheinen bei der Nutzung von Web-Anwendungen eine Abwägung zwischen *vernünftig und sicher* vs. *impulsiv und spaßig* zu treffen, wobei die Bedeutung des Spaßes bei der Nutzung einer Anwendung nicht unterschätzt werden sollte. Er darf zu Gunsten einer erhöhten Sicherheit nicht zu sehr leiden.

«*Sie solle einen ähnlichen Funktionsumfang anbieten wie ...*» Natürlich kennen und schätzen viele Benutzer bereits aktuelle Web-Anwendungen. Dort organisieren sie aktuell ihre Online-Kontakte oder die Anwendungen sind bereits Teil ihres Arbeitsworkflows. Eine Anwendung, die auf einer alternativen Architektur aufbaut, sollte einen ähnlichen Funktionsumfang zur Verfügung stellen können, wie er aus vielen aktuellen Anwendungen zum Teilen eigener Inhalte bekannt ist.

«*Sie solle ein Community-Projekt sein und nicht durch eine zentrale Instanz betrieben/entwickelt werden*» Diese Aussage scheint ebenfalls mit der Vetrauensbildung in Beziehung

zu stehen. Es ist anzunehmen, dass ein Projekt, das von einer globalen Community entwickelt wird, dieser Person vetrauenswürdiger erscheint als eines, welches durch ein wirtschaftlich handelndes Unternehmen vorangetrieben wird.

«*Ich möchte meine aktuellen Inhalte mit der neuen Architektur nutzen und neue, dort erstellte Inhalte auch wieder woanders hin mitnehmen können.*» Viele Benutzer haben aktuell schon eigene Inhalte im Web veröffentlicht und Kontakte geknüpft. Eine Berücksichtigung bereits existierender Inhalte erscheint sinnvoll bei der Entwicklung der neuen Architektur und darüber hinaus sollten Inhalte, die mit Unterstützung der neuen Architektur entstehen auch nicht in dieser verharren müssen. Es sollte kein *walled garden* entstehen, der dem Gedanken von *Data Portability* im Web entgegen steht.

«*Ich möchte weiterhin die Social Networks X und Y nutzen, können diese nicht einfach aufgerüstet werden, um Vertraulichkeit, Verfügbarkeit und Integrität meiner Inhalte zu gewährleisten?*» Die Bedeutung dieses Wunsches ist nicht zu unterschätzen: Natürlich nutzt ein Großteil der befragten Personen bereits jetzt zahlreiche Social Networks und hat dort nicht nur schon viele Inhalte hinterlegt, sondern auch seine Kontakte, den *Social Graph*, versammelt und eine Reputation erlangt. Dies sind aktiv betriebene Kommunikationskanäle, die sicherlich wenige bereit sind aufzugeben für die Nutzung einer Anwendung, die auf einer neuartigen Architektur aufbaut, jedoch in der Anfangsphase gänzlich unbekannt sein wird und kaum weit verbreitet. Bei komplett voneinander isolierten Systemen (Anwendung mit neuartiger Architektur vs. bekanntes Social Network) verlieren Benutzer, die sich für die Nutzung der neuen Anwendung entscheiden all ihre Kontakte, die sich nicht dafür entscheiden. Es sollte aus diesem Grund im Rahmen der Anforderungsermittlung überlegt werden, inwiefern auch aktuelle Social Networks, die die Gewährleistung der Schutzziele nicht erfüllen in einen Kontext integriert werden können, der eben diese Gewährleistung für seine Benutzer ermöglicht, damit jeder Benutzer für sich entscheiden kann, welches System er mit wem zum Teilen seiner Inhalte nutzen möchte.

2.2.3 Ökosystem

Die Inhalte der Benutzerbefragungen ergaben wertvolle Erkenntnisse und Impulse hinsichtlich weiterer Recherchen. In den folgenden Abschnitten soll nun zu einzelnen Aspekten und Fragestellungen ein Status-Quo erarbeitet und damit das Ökosystem beschrieben werden, in das sich eine neue Architektur einbetten würde.

2.2.3.1 Ausgewählte Aspekte der Privatsphäre im Web

Inhalte von Menschen im Web vertraulich zu behandeln, das bedeutet ihre Privatsphäre zu wahren. Das Konzept der Privatsphäre wird im Web kontrovers diskutiert und ebenso verschieden gelebt[18]. Ob es die Offenheit des Webs ist, die potentielle Anonymität, das Spielen mit Pseudonymen oder die Möglichkeit viele Menschen auf einmal zu erreichen und dieser Reiz, der davon ausgeht, auf eigene Handlungen viele Reaktionen zu erfahren — die genauen Ursachen sind sicherlich vielschichtig, jedoch ist beobachtbar, dass Menschen im Web ihre im Offline-Leben meist bewusst gewahrte Privatsphäre mitunter etwas lockerer handhaben und viele persönliche Informationen über sich veröffentlichen. Öffentlich zugängliche Daten über identifizierbare Menschen sind wirtschaftlich natürlich attraktiv (z.B. für das Marketing) und diese Tatsache ist sicherlich auch *ein* Motor bei der gegenwärtigen Bildung einer neuen offenen Privatsphäre-Kultur im Web: Nicht das Verbergen von persönlichen Informationen wird propagiert, sondern der „bewusste, aber öffentliche Umgang“ mit Informationen zur eigenen Identität. Man solle ein möglichst authentisches Bild im Web von sich erschaffen und auffindbar sein; wer im Web unauffindbar ist, der habe schlechte Chancen bei Vorstellungsgesprächen, da ihm Paranoia, Technik- oder Fortschrittsverweigerung vorgeworfen würden — diese Sichtweise wird landläufig als Tatsache angenommen und ist in dieser Form auch regelmäßig in Medienberichten zu erkennen. Natürlich scheint diese Perspektive etwas eingeschränkt, kann jedoch auch nicht völlig von der Hand gewiesen werden. Es entsteht im Web ein Handlungsdruck sich mit der Veröffentlichung von Informationen über die eigene Person zu beschäftigen und eine eigene Handhabe dafür zu finden.

Manche Web-Anwendungen nehmen hinsichtlich der Privatsphäre ihrer Benutzer extreme Perspektiven an. Die Fotografie-Anwendung *Instagram*[19] beispielsweise, die für die iOS-Plattform von Apple entwickelt wird und mit der sich unterwegs aufgenommene Fotos im Web teilen lassen, beinhaltet den in Abbildung 2.3 dargestellten Dialog: Das standardgemäße vollkommen öffentlche Teilen der Fotos unter allen Instagram-Benutzern und die Auffindbarkeit im Instagram-Verzeichnis kann nachträglich deaktiviert werden, allerdings nur durch die Bestätigung einer Warnmeldung, dass man kurz davor sei die eigene Privatsphäre zu aktivieren und dies vor allem Nachteile mit sich zöge — eine aus Datenschützersicht absurd erscheinende Warnung.

Ein allgemein verbreitetes Bild der eigenen Privatsphäre im Web veranschaulicht überspitzt auch Abbildung 2.4. Das Internet und die Privatsphäre seien zwei Kon-

[18] genauer formuliert besitzen nicht Menschen eine Privatsphäre im Web, sondern deren Personae. Diese können die Privatsphäre dabei unterschiedlich offen interpretieren.

[19] http://instagr.am/

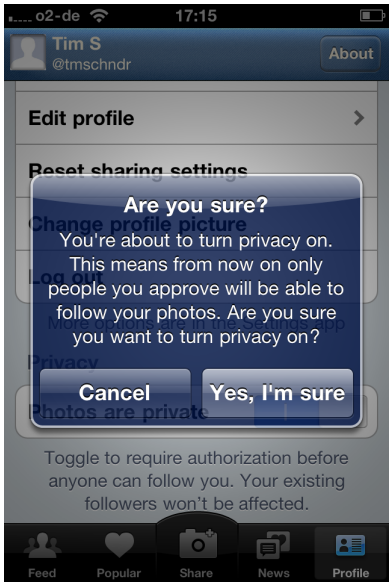

Abbildung 2.3: Dialog der iOS-Software Instagram: „Achtung, Sie sind dabei ihre Privatsphäre zu aktivieren!“.

zepte, die zueinander finden könnten. Der Autor des Diagramms kommentiert: «*If you don't want to see it spreading on the internet, don't put it there at all.*» Ist das Web also ein Medium, in dem eigene Inhalte per se unkontrollierbar sind und aus dem man private Inhalte komplett heraushalten sollte?

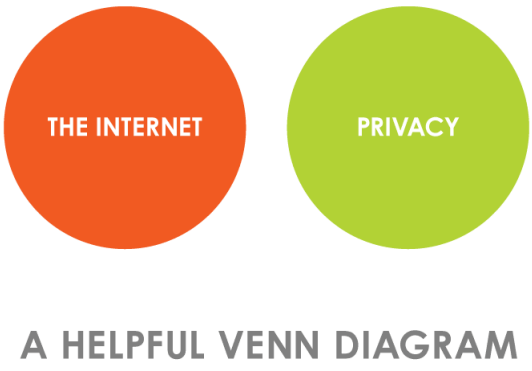

Abbildung 2.4: Eine oft vertretende Meinung visualisiert in einem Mengendiagramm: Das Internet und die Privatsphäre bilden keine Schnittmenge.

Dies muss nicht die notwendige Konsequenz sein, sondern es sollte Benutzern Möglichkeiten gegebenen werden ihre im Web veröffentlichten Daten besser zu organisieren, zu kontrollieren und nachzuvollziehen was mit ihnen genau geschieht. Diese Meinung vetritt die Initiative *Mozilla Privacy Icons* der Mozilla Foundation, die

Aufklärung leisten möchte was genau mit Benutzerdaten im Web geschieht und dazu eine leicht verständliche Alternative zu den üblichen in „Juristendeutsch“ formulierten Datenschutzerklärungen von Web-Anwendungen einführen möchte — ähnlich wie es die *Creative Commons*-Initiative geschafft hat, einfach Symbole für Lizenzvereinbarungen in der Web-Landschaft zu verankern [Ras10b]. Eine Arbeitsgruppe hat dazu sieben relevante Aspekte zur Nutzung eigener Inhalte im Web erarbeitet, deren Ausprägung pro Web-Anwendung per Symbol gekennzeichnet werden sollte [Ras10a]:

- **Is your data used for secondary use?** Teilt eine Web-Anwendung die Inhalte ihrer Nutzer mit Dritten, sollten diese davon in Kenntnis gesetzt werden.
- **Is your data bartered?** Schlägt eine Anwendung Profit aus einem Tausch dieser Inhalte mit Dritten, sollten die Nutzer auch darüber informiert werden.
- **Under what terms is your data shared with the government and with law enforcement?** Unter welchen Umständigen gibt ein Unternehmen auf staatlichen Druck die Inhalte eigener Nutzer oder weitere persönliche Informationen über seine Nutzer heraus? Gerade bei File-Sharing-Netzwerken, die zum illegalen Austausch von Multimedia-Dateien genutzt werden können, kann diese Angabe für Nutzer relevant sein.
- **Does the company take reasonable measures to protect your data in all phases of collection and storage.** Es gibt zahlreiche Wege Inhalte persistent zu speichern oder während der Übertragungen abzusichern. Wie genau die Daten vorgehalten werden und wer Zugriff auf sie besitzt, sollte durch ein weiteres Icon beschrieben werden.
- **Does the service give you control of your data?** Können Daten gelöscht werden, wenn ein Benutzer dies verlangt? Können sie nachträglich verändert werden? Welche Art von Kontrolle kann ein Benutzer tatsächlich ausüben?
- **Does the service use your data to build and save a profile for non-primary use?** Diese Frage zielt for allem auf Daten, die unmittelbar mit einem bestimmten Benutzern in Verbindung gebracht werden können. Ob auach diese ausgewertet werden, um ein internes Profil einer Person zu entwickeln, das im Weiteren z.B. für Marketingzwecke genutzt werden kann, soll für einen Benutzer deutlich werden.
- **Are ad networks being used and under what terms?** Viele Web-Anwendungen haben Werbung anderer Werbedienstleister integriert und reichen zur Abrechnung Statistiken über die Nutzung und das Verhalten der

eigenen Benutzer an diese Unternehmen weiter. Auch in diesem speziellen Punkt sollte Transparenz für den Benutzer herrschen.

Einen ersten Entwurf der Privacy Icons zeigt Abbildung 2.5. Das Projekt befindet sich aktuell noch in der Frühphase, die Initiative ist jedoch aus den bisher in dieser Ausarbeitung gewonnen Erkenntnissen heraus positiv zu bewerten — wenn sie auch ein Problem noch zu bewältigen hat: Privacy Icons mögen Datenschutzerklärungen visualisieren und dadurch Transparenz und Vertrauen schaffen, doch eine Gewissheit, ob die getätigten Aussagen tatsächlich wahr sind, sie kann wohl in den meisten Fällen nicht erlangt werden.

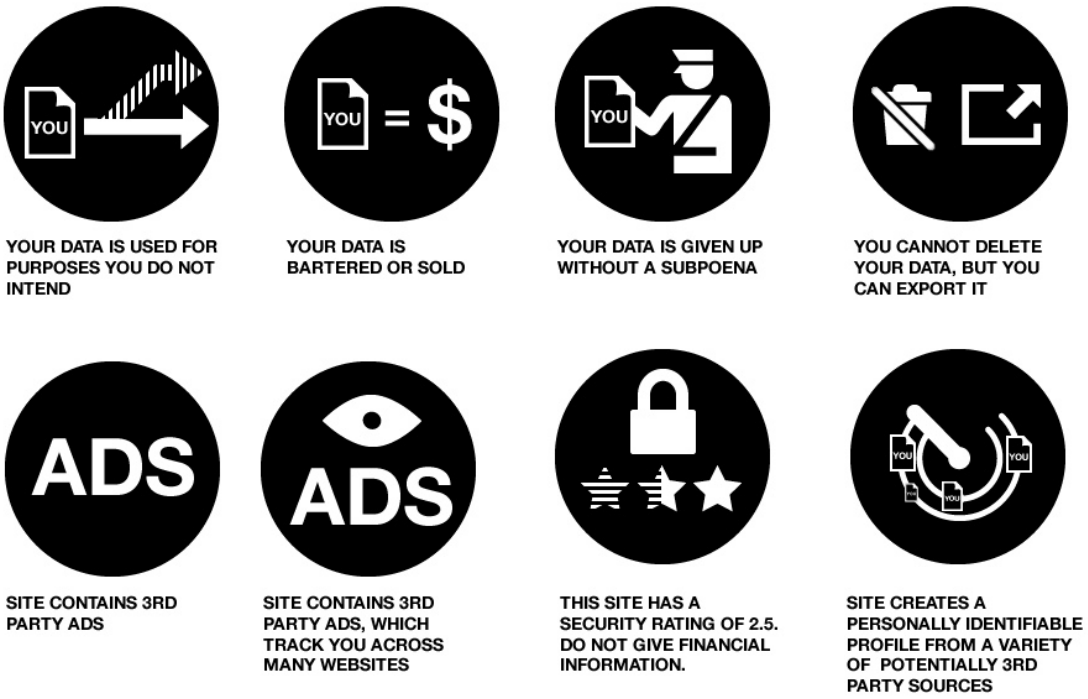

Abbildung 2.5: Mozilla Privacy Icons sollen helfen die Nutzung eigener Daten in Web-Anwendungen transparenter zu gestalten.

2.2.3.2 Vertrauenswürdigkeit im Web

Das Projekt Mozialla Privacy Icons zeigt es gut: Der Ansatz, ein Konzept der Privatsphäre im Web zu etablieren ist sinnvoll, ebenso es so nachvollziehbar wie möglich zu gestalten, was genau mit eigenen Daten im Web passiert. Doch muss man auch hier den genutzten Icons vertrauen, die Web-Anwendung muss vertrauenswürdig erscheinen. Erst dann kann man diesen Angaben tatsächlich glauben schenken.

Je mehr Inhalte eigene eine Person in Web-Anwendungen organisiert, desto relevanzter wird für diese der Aspekt der Vertrauenswürdigkeit, auch wenn sie (wirtschaftlich motiviert) bisher eher im Bereich des Online-Shoppings von Nöten war. In diesem Umfeld findet man daher derzeit schon Bemühungen Vertrauenswürdigkeit zu erwecken. Früh war im Web klar: Ein Online-Einkauf benötigt unbedingt eine Vertrauensbeziehung zwischen den Geschäftspartnern. Denn die Ware wird nicht direkt händisch übergeben, sondern man erwartet, dass sie in der angekündigten Zeit zugeschickt wird. Um an Vertrauenswürdigkeit zu gewinnen, beschreiben Online-Shops daher z.B. ihre Infrastruktur und ihre Dienstleister, machen Angaben zu Logistikpartnern und Abrechnungsunternehmen. Man findet viele Bemühungen an Transparenz zu gewinnen. Darüber hinaus lassen sich viele Shops zudem mit Gütesiegeln oder Sicherheitszertifikaten ausstatten. Diese extern vorgenommenen Prüfungen der Datensicherheit und Bonität sollen Käufern als Grundlage dienen eine Vertrauensbeziehug aufzubauen[20].

Als auch Privatpersonen ebenfalls anfingen Produkte über das Web zum Verkauf anzubieten (z.B. bei Online-Auktionshäusern wie ebay[21] oder im Amazon Marketplace[22]), wurden neue Möglichkeiten geschaffen Vertrauenswürdigkeit für Privatpersonen zu modellieren: Der Begriff der *Online-Reputation* entstand. Bei ebay sind es Bewertungen anderer Käufer, aus der sich ein „Zuverlässigkeits-Score“ errechnen lässt. Wissenschaftliche Untersuchen zeigen vielfach, dass die ebay-Reputation ausschlaggebend für die tatsächliche Kaufentscheidung ist und eine wichtige Grundlage bei der Auswahl eines Verkäufers, wenn ein Produkt mehrfach zum gleichen Preis angeboten wird (vgl. [HW01] [RZSL06]). Als ebay-Nutzer auch bei Aktivitäten außerhalb ebays zur Kenntlichmachung ihrer Online-Reputation auf ihre dort dokumentierte positive Bewertung verwiesen, erkannte man, dass die Reputation als wichtiger Teil der Online-Identität eines Menschen nicht in einem abgeschlossenen System verweilen sollte und es bildeten sich Dienste, die eine Web-weite Reputation zur Verfügung stellen möchten[23]. Bei diesen kann man sich für beliebige Aktionen, Handlungen oder einfach persönlichen Eigenschaften von anderen Benutzern bewerten lassen.

Die Meinung von prominenten Personen, denen allein aufgrund ihrer allgemeinen Bekanntheit eine Glaubwürdigkeit anhaftet, wird weiterhin gerne in Form so genannter *Testimonials* auf Werbeseiten für Web-Produkte platziert. Diese Personen berichten

[20] dies geht zusammen mit der Transparenz als Voraussetzung für Vertrauenswürdigkeit ebenfalls einher mit den Erkenntnissen der Benutzerinterviews: Externe Autoritäten ermöglichen einen Vertrauensaufbau.

[21] http://www.ebay.de/

[22] http://www.amazon.de/gp/help/customer/display.html/ref=hp_ln_amp/?nodeId=886416

[23] z.B. RapLeaf http://www.rapleaf.com/

dann meist aus ihrer positiven Erfahrung während der Nutzung eines Produktes (z.B. eines Social Networks) und wie z.B. eines ihrer Probleme dadurch gelöst werden konnte. Entdecktr der interessierte Benutzer dann bei sich ein ähnliches ungelöstes Problem und schenkt diesen Testimonials glauben, dann gewinnt das Produkt an vertrauenswürdigkeit und eine Nutzungsmotivation kann die Folge sein.

Spricht man aktuell von Vertrauen-schaffenden Maßnahmen im Web, so finden als Urteile externer Autoritäten in erster Linie Testimonials bekannter (Web-)Persönlichkeiten, Gütesiegel oder Prüfplaketten Verwendung, letztere zur Verdeutlichung der Datensicherheit auch immer in in Social Networks. Auf Benutzer-Ebene sind es Reputations-Bewertungssysteme, die mitunter parallel in geschlossenen, separaten Systemen existieren und bisher nur selten vereinzelt verwendbar erprobt werden.

2.2.3.3 Alternative, dezentrale Web-Anwendungen

Wie in der Heranführung an das Thema der Ausarbeitung dargelegt, fußt das Problem vieler Web-Anwendungen auf einer zentralistisch organisierten Architektur. Es erscheint daher zunächst eine kluge Idee, dass die zu entwerfende Architektur dezentral aufgebaut sein sollte, also ohne zentrale Anwendungslogik, Datenhaltung und vor allem ohne einen übermächtigen Betreiber, der sämtliche Systeme kontrolliert. Völlig neu ist diese Idee allerdings nicht. Für viele populäre Web-Anwendungen existieren mittlerweile auch dezentrale Alternativen. Denn mit jeder wachsenden Popularität einer zentral organisierten Web-Anwendung beginnen in der Regel kleine oder größere Initiativen diese Anwendung in dezentraler Form nachzubilden, um damit einzelne Schutzziele der IT-Sicherheit bei gleichem Funktionsumfang und gleicher Freude an der Benutzung zu gewährleisten.

sparkleshare vs. Dropbox Der amerikanische File-Sharing-Dienst *Dropbox* erfreut sich derzeit großer Beliebtheit im Web und gab bereits früh im Jahr 2010 bekannt, die 4-Millionen-Nutzer-Grenze überschritten zu haben[24]. Dropbox ermöglicht es ganze Bereiche der eigenen Festplatte mit Kontakten zu teilen. Dateien, die in diesen Ordnern abgelegt werden, werden automatisch ins Web und auf die Endgeräte der Kontakte synchronisiert (derzeit unterstützt werden Windows-, Linux- und Mac OS X-Systeme, Apple iOS-Systeme und Google Android-Systeme). Zwar geschieht die Kommunikation und Datenspeicherung dabei nach Angaben des Unternehmens stets verschlüsselt[25], dennoch bleiben die Daten natürlich in der Kontrolle

[24] vgl. [Wau10]
[25] vgl. [Dro10]

des zentralen Dienstes, eine Verfügbarkeit ist nicht gesichert. Aus dieser Tatsache heraus hat sich das Projekt *sparkleshare*[26] gegründet, welches ein ähnliches Teilen von Dateien auf einer selbst-kontrollierten Architektur anstrebt. Statt der Dropbox-Server-Landschaft sollen dann eigene Rechnerressourcen zusammen mit einer ähnlichen Client-Anwendung genutzt werden können. Am 14. August 2010 wurde das zweite Alpha Release eines Linux-Clients für sparkleshare veröffentlicht, am 5. September eine erste Beta-Version. Seitdem herrscht bis heute [Stand: Ende Dezember 2010] Stillstand, während Dropbox mit enormen Wachstumsraten wirbt.

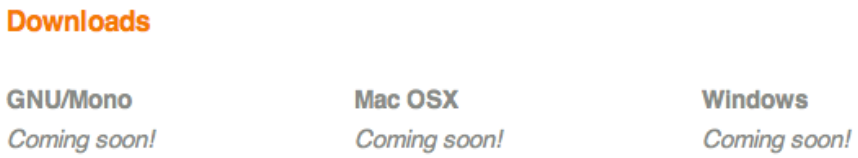

Abbildung 2.6: Im Downloadbereich der Dropbox-Alternative *sparkleshare* wird derzeit noch nicht viel angeboten.

Eine freie Implementierung zu einem Produkt eines erfolgreich wirtschaftlich arbeitenden Unternehmens im Markt zu etablieren, scheint als äußerst schwieriges Unterfangen. Nicht nur hat man mit Marketingproblemen zu kämpfen, meist fehlen auch Investitionen in die Entwicklung. Der Aufwand mit einem gut arbeitenden Konkurrenten auf Augenhöhe zu bleiben ist durch ehrenamtliche Arbeit meist nicht zu bewerkstelligen. Projekte haben es schwer mit dem Funktionsumfang bekannter Software mitzuhalten und dabei eine ähnlich einfache Handhabung der Sofwtare zu ermöglichen, während sie schließlich auf wesentlich komplexerer, weil verteilter Architektur aufsetzen. Bei sparkleshare müssen sich Benutzer dadurch auch um die Administration der Software kümmern. Nicht selten verbleiben derartige Ansätze daher in Kreisen ausgebildeter Informatiker und anderer Fachleute. Im Web wird in diesem Zusammenhang daher auch der Begriff des *Nerd Social Network* benutzt, um die primäre Zielgruppe zu pointieren.

StatusNet vs. Twitter Anders ist es dem Projekt *StatusNet* ergangen, welches sich zunächst als offene Implementierung zu Twitter positionierte: Ein quell-offener Microblogging-Dienst, der auf eigener Infrastruktur betrieben werden konnte, dies war die erste Projektskizze des in der Anfangsphase noch als *laconica* bezeichneten Projektes[27]. Mit einer Referenzimplementierung unter dem Namen *identi.ca*[28] konnten Interessierte von der Funktionalität und den Vorzügen einer freien Implementierung

[26] http://sparkleshare.org/
[27] http://status.net/
[28] http://identi.ca/

— im Wesentlichen: die bessere Kontrolle der eigenen Inhalte — überzeugt werden. Zwar weist identi.ca aktuell eine wesentlich kleinere Nutzerzahl auf, als der direkte Konkurrent Twitter, doch entwickelte das Projektteam die Idee „Microblogging“ weiter: die StatusNet-Implementierung ermöglicht nun nicht mehr nur eine eigene Installation eines Microblogging-Services, sondern erlaubt auch die gemeinsame Nutzung vieler dezentraler StatusNet-Installationen über ein gemeinsames Protokoll[29] und mittlerweile ist aus dem Community-Projekt ein Unternehmen herausgewachsen, welches durch solides Investment aus der Wirtschaft gestützt die Entwicklung der freien Software hauptsächlich weiterführt und darüber hinaus auch ein Hosting der Software anbietet[30]. Zielgruppe sind vor allem Unternehmen, die ein internes Microblogging-System nutzen möchten.

Bei diesem recht einfachem Anwendungsfall „Microblogging“ konnte eine freie Software erschaffen werden, die den sehr begrenzten Funktionsumfang von Twitter offenbar zufriedenstellend nachbilden konnte. Durch eine Referenzimplementierung als SaaS haben Interessierte die Möglichkeit die Software unmittelbar auszuprobieren. Zwar bleibt Twitter weiterhin Urgestein und Platzhirsch im Microblogging-Segment des Webs, doch existiert mit StatusNet nun eine mächtige und teilweise auch mächtigere freie Implementierung, die im Gegensatz zum Vorbild einen gänzlich neuen Anwendungsfall, nämliche die interne, geschlossene Nutzung von Microblogging (z.B. im Unternehmskontext) unterstützt und entsprechend vermarktet wird.

diaspora vs. Facebook Eine große Aufmerksamkeit im Bereich *Distributed Social Networking* konnte jüngst das studentische Projekt *diaspora** auf sich ziehen[31]. Im Juni veröffentlichte das New Yorker Team aus vier Informatikstudenten einen Entwurf eines verteilten Social Networks und bat die Web-Gemeinde daraufhin um 10.000 US-Dollar Startkapital, um im Sommer 2010 ihre Alternative zu Facebook entwickeln zu können. Da Facebook zu dieser Zeit aufgrund seiner undurchsichtigen Privatsphäre-Einstellungen öffentlich in der Kritik war, wurde die Idee einer dezentralen Social Network-Alternative in den Medien gerne diskutiert und das diaspora-Team erhielt wesentlich mehr Spenden als erwartet. Ende Mai begannen die Entwicklungen an der Software, die schlussendlich mit über 200.000 US-Dollar gespendeten Budget finanziert werden konnte[32]. Anschließend wurde es während der Entwicklung etwas stiller um das Projekt, aber die Erwartungen der Web-Öffentlichkeit verblieben nach diesem bemerkenswerten Start extrem hoch. Eine erste Veröffentlichung des Source

[29] Auf die Besonderheit des benutzten Protokolls *OStatus* wird in einem folgenden Abschnitt näher eingegangen.
[30] vgl. [Sch10b]
[31] https://joindiaspora.com/
[32] vgl. [Sie10a]

Codes wurde für Mitte September angekündigt und fand schließlich am 15. Oktober auch statt: Die Erwartungen konnten jedoch nicht erfüllt werden und es machte sich zum Teil große Enttäuschung in der Weblog-Landschaft breit[33].

Die Architektur, die diaspora verfolgt, weist im Wesentlichen die Struktur eines Peer-2-Peer-Netzes auf: Dezentral betriebene Server, bei diaspora *pods* genannt, organisieren die lokale Datenspeicherung und -verteilung im Netzwerk. Benutzer können entweder selbst einen eigenen pod betreiben, der Source-Code dafür steht zum Download bereit, oder sich einen Account auf einem öffentlichen pod erstellen, den jemand anderes betreibt. Während diese Herangehensweise allgemein akzeptiert wurde, war es die Art der Implementierung, die weniger Begeisterung fand. Die Code-Qualität überzeugte beim ersten Release kaum und die Software wies teils fatale Sicherheitslücken auf, die gerade in Hinblick auf das Ziel, vertrauliche Kommunikation zu ermöglichen, umso mehr verwunderten. Falk Hedemann kommentiert im t3n Magazin [Hed10]:

«*Lange hat es nicht gedauert, bis die ersten sicherheitsrelevanten Lücken aufgedeckt wurden. Wie The Register meldet, haben Experten bereits Möglichkeiten entdeckt fremde Accounts zu übernehmen, ohne Erlaubnis neue Kontakte aufzubauen oder Fotos zu löschen. Entwickler haben sich den Code bereits genauer angesehen und sind enttäuscht: „Diaspora ist eine einfache Rails App, mit der man Fotos hochladen kann“, zitiert Mashable den Entwickler J. Chris Anderson. Daraus könne man schließen, dass die Codebasis keinesfalls ausreicht, um daraus in den nächsten Monaten einen echten Konkurrenten für Facebook zu machen.*»

Seitdem konnte durch die öffentliche Beteiligung der Community an der Entwicklung von diaspora zwar an der Code-Qualität gearbeitet werden, so dass nun Ende Dezember tatsächlich vereinzelte pods zu Testzwecken betrieben werden, doch ist das Projekt noch weit entfernt von einer Marktfreife für die Masse an Web-Nutzern. Denn zeigt sich bei diaspora noch deutlicher als bei anderen Ansätzen ein generelles Problem bei der Entwicklung neuartiger Software:

Das diaspora-Team wollte offensichtlich neueste Web-Techniken nutzen, um es Benutzern zu ermöglichen die Anwendung so universal wie möglich nutzen zu können. Aus diesem Grunde findet statt einer relationalen Datenbank die Dokument-basierte Datenbank *MongoDB*[34] Verwendung. Diese wird angesprochen über ein Backend, das in der Programmiersprache *Ruby*[35] unter Zuhilfenahme des Frameworks *Rails*[36] geschrieben wurde. Um nun einen eigenen pod zu betreiben — was schließlich die Idealvorstellung sein sollte, da genau dann jeder Benutzer die *alleinige* Kontrolle

[33] vgl. [Sie10b]
[34] http://www.mongodb.org/
[35] http://www.ruby-lang.org/de/
[36] http://rubyonrails.org/

über seine eigenen Inhalte besitzt und nicht der Betreiber eines öffentlichen pods, der zwar nicht Facebook ist, aber trotzdem aus Sicht des Benutzers eine Dritte Person bleibt — muss nun ein (Linux-)Server eingerichtet werden, auf dem eine *Ruby on Rails*-Umgebung installiert wird und ein Master-Datenbanksystem *MongoDB*, welches aufgrund seiner absichtlich nicht garantierten Zuverlässigkeit mindestens einmal in ein Slave-System repliziert werden sollte. Dies ist keine Aufgabe, die man einem normalen Endnutzer im Web stellen sollte. Denn das Hosting von Ruby-Anwendungen im Web stellt auch zur Zeit immer noch besondere Anforderungen an Server-Umgebungen und Administration und wird von nur wenigen Unternehmen im *Consumer*-Segment angeboten. PHP-Umgebunden für Web-Anwendungen hingegen erhält man vorinstalliert bei vielen großen Webhosting-Unternehmen für wenig Geld und diese sind, z.B. für das Betreiben eigener Weblog-Software, aktuell auch schon weit verbreitet.

Wenn es also tatsächlich das Ziel von diaspora ist, eine alternative Software zu Facebook zu schaffen, d.h. die gleiche Zielgruppe zu adressieren, die Facebook aktuell bedient, dann ist eine Auswahl derartiger Systemkomponenten das Ergebnis einer misslungenden Anforderungsermittlung und schlicht an der Zielgruppe vorbei implementiert. Trotz des Eifers und einem prinzipiell gutem Vorhaben bleibt der Spott von Kritikern und die Nachsage das Thema nur halbherzig aufgearbeitet zu haben. Diese Annahme offenbart sich an anderer Stelle auch in der Art und Weise des Austausches zwischen diaspora pods: Hier wird nicht auf einen Webstandard zurückgegriffen, wie ihn *OStatus* oder *OAuth* darstellen könnten[37], sodern die Implementierung erfolgt proprietär — nicht im Sinne von *closed source*, sondern als zwar quell-offene, aber dennoch selbst entworfene Art der Kommunikation. diaspora läuft so Gefahr zur Insellösung zu werden: Innerhalb des Netzwerkes kann dann möglicherweise vertraulich kommuniziert werden, doch wenn durch hohe Anforderungen an das Technikverständnis weder eine breite Masse an Benutzern erreicht werden kann, noch die Verbindung zu anderen Social Networks aufgrund proprietärer Kommunikationswege besteht, dann kann die Problemlösung nicht zufriedenstellend sein.

Geeky NoseRub Unter einem ähnlichen Problem scheint *NoseRub*[38] zu leiden, ein interessanter, deutscher Ansatz zur Schaffung eines dezentralen Social Networks. NoseRub stellt dabei allerdings kein eigenes, geschlossenes System dar, sondern ermöglicht eher auf Protokollebene die Kommunikation zwischen bestehenden Social Networks und erlaubt das dezentrale Teilen über Social Network-Grenzen hinweg,

[37] auf beide Ansätze wird in einem späteren Abschnitt näher eingeganen
[38] http://noserub.com/

indem auf das eigene NoseRub-System auch eigene Inhalte aggregiert werden, die in anderen NoseRub-fähigen Social Networks erzeugt wurden.

Der Claim auf der NoseRub-Webseite dazu lautet:

«*NoseRub wants to be inspiration, protocol and implementation of a decentralized social network. Sounds geeky? It surely is...*»

Martin Weigert kommentiert diesen als Autor des Technik-Weblogs netzwertig.com folgendermaßen [Wei10]:

«*Ein dezentrales Social Network hat nur dann Chancen auf Massenerfolg, wenn es den Machern gelingt, den „Geek-Faktor" und die technischen Einstiegshürden so minimal wie möglich zu halten. Offen dargestellter Stolz darüber, wie „geeky" NoseRub ist, sind da eher kontraproduktiv, auch wenn es sich um ein Protokoll und nicht um einen Dienst für den Endanwender handelt.*»

Bereits das Auftreten alternativer Software und dessen Wahrnehmung bei potentiellen Nutzern sollte daher nicht unterschätzt werden. Gerade hinsichtlich der Vertrauenswürdigkeit einer Architektur kann dies unerfahrene Nutzer verschrecken und bei ihnen den Eindruck erwecken, dass sie jene Software wohl nicht verstehen — selbst wenn sie sich damit beschäftigen würden.

2.2.3.4 Zusammenfassung

Zusammenfassend lässt sich für die Betrachtung des „Ökosystems" festhalten:

- Zwar wird die *Privatsphäre* eines Menschen im Web teilweise vollkommen anders interpretiert und gelebt, als man es aus dem Offline-Kontext her gewohnt ist, doch existieren trotzdem Bemühungen und Bedürfnisse eine digitale Privatsphäre für Personae im Web global zu etablieren.
- Das Thema *Vertrauenswürdigkeit im Web* trat in der Frühphase des Webs zunächst eher im wirtschaftlichen Kontext Online-Shopping auf, wird jedoch mit dem Wachstum von Social Networks und der Vermehrung nutzergenerierter Inhalte immer mehr auch in diesen Nutzungskontexten aktuell. Das Negativbeispiel StudiVZ zeigt gut, wie Vertrauenswürdigkeit durch die Menschen hinter einem Projekt trotz mittlerweile durchweg positiver externer Bewertungen „verspielt" werden kann.
- *Alternative, dezentrale Web-Anwendungen* sind keine Neuigkeit im Web. Seit Jahren existieren immer wieder Bestrebungen dezentrale Alternativen für populäre Web-Anwendungen zu etablieren. Alle Ansätze kämpfen dabei jedoch

mit dem Problem, dass eine dezentrale Architektur meist komplexer wird als die des Originals und von seinen Nutzern vielfach zusätzlich zur Nutzung auch Installation, Administration und dabei Web-Fachwissen abverlangt. Ein Leser kommentierte einen Artikel im netzwertig.com-Weblog über dezentrale Social Networks mit der treffenden Meinung [Wei10]: *«Ich glaube (...), dass sich keine offene Alternative durchsetzen wird, solange man nicht auf der Einstiegseite einen großen „Join now" Button findet.»* Auch aus den Erkenntnissen der Befragungen wird deutlich: Ein Großteil der Benutzer wird nicht bereit sein, manuell in vielen Schritten einen eigenen Server einzurichten. Auch werden sich Benutzer nicht mit weniger Funktionalitäten zufrieden geben, als sie es heute von den großen Social Networks gewöhnt sind. Trotzdem lässt sich nicht zuletzt durch diaspora* ein eindeutiger Trend in Richtung Peer-2-Peer-Social Networking zu beobachten. Allein die passende Umsetzung scheint derzeit noch nicht gefunden.

2.3 Modeling

Nach der umfassenden Recherchephase schließt sich im Goal-directed Design-Prozess nun die Modellierungsphase (Modeling) an. Die gewonnen Erkenntnisse sollen in strukturierte Modelle überführt werden, die im weiteren Entwicklungsprozess einfacher genutzt werden können als die umfangreichen Aufzeichnungen der Benutzer-Interviews und die übrigen Recherchepapiere.

2.3.1 Personas

Auf Basis der Interviewergebnisse werden daher nun im Rahmen einer Benutzermodellierung *Personas* erstellt. Personas sind deskriptive Modelle potentieller Nutzer. Sie sind aus qualitativer Recherche komponierte Archetypen und beschreiben wie sich einzelne Benutzer verhalten, denken, was sie durch die Nutzung eines Systems erreichen möchten und warum. Personas sind allerdings keine direkten Abbildungen einzelner Interviewpartner, aber sie basieren auf deren beobachteten Verhaltensmustern, Motivationen und Sichtweisen und bilden diese in einen bestimmten Nutzungskontext ab. Damit Personas effektiv im Designprozess genutzt werden können, gilt es die relevanten und bedeutsamen Verhaltensmuster in der Zielgruppe zu erkennen und diese als Basis für die Erstellung von Personas zu nutzen. Im Rahmen der „Persona Hypothese“ (Abschnitt 2.2.2.1) wurden dazu Dimensionen von Verhaltensweisen beschrieben, aus deren tatsächlichen Ausprägungen sich nun Muster erkennen lassen. Im Rahmen dieser Benutzermodellierung sind dies:

- der Extravertierte, der gerne öffentlich Inhalte teilt und alle komfortablen und bequemen Vorzüge des Webs vollständig auskostet, aber dennoch Integrität, Verfügbarkeit und Authentizität seiner Inhalte schätzt.
- der Domänenexperte, der alle Techniken, Risiken und Konsequenzen seines Handelns im Web einschätzen kann und auf dieser Wissensbasis neue Architekturen beurteilt, die er vor allem zur beruflichen/professionellen Kommunikation einsetzen möchte.
- der Skeptiker, der absolute Privatsphäre im Web wünscht und sich aufgrund seines Grundmisstrauens gegenüber allen wirtschaftlich betriebenen Social Networks im Web nur vorsichtig bewegen kann und zu Gunsten des Schutzes seiner Privatsphäre und Identität gerne Kompromisse hinsichtlich des *Joy of Use* eingeht.
- der Unbedarfte, der ohne großes Fachwissen einfach das Web als tolle Plattform

entdeckt hat, über die er mit Freunden in Kontakt bleiben kann. Ohne sich groß Gedanken zu machen, verteilt er seine Inhalte im Web und ist sich, solange dies funktioniert, keines Problems bewusst.

- der Unerfahrene, der, ausgestattet mit dem Basiswissen wie ein Computer und ein Browser zu bedienen ist, zögerlich, aber durch sein Umfeld motiviert, das Teilen eigener Inhalte im Web ausprobiert.

2.3.1.1 Michel „muck“ Langhanns

Kurzbeschreibung/Verhaltensmuster Michel verkörpert den extravertierten Benutzertypen, der das Web intensiv und regelmäßig nutzt und dabei sein Leben täglich online dokumentiert. Das Konzept der Privatsphäre kennt er, aber online lebt sein Alter-Ego „muck“ vollständig öffentlich und wird privat manchmal mit ihm verwechselt.

Michel ist 31 Jahre alt und arbeitet als Grafikdesigner in einer Web-Agentur in Hannover. Bei seinem bürgerlichen Namen nennen ihn allerdings nur alte Schulfreunde und seine Eltern. Für die meisten seiner Kontakte hingegen, die er zum überwiegenden Teil aus dem Online-Kontext kennt, ist er schlicht „muck“. Sein Name ist seine Marke. Schon früh in den 1990er Jahren der *New Economy* entdeckte muck das Web und machte es zu seinem neuen Arbeitsumfeld. Seitdem entwirft Michel in der von ihm mitgegründeten Agentur visuelle Benutzungsoberflächen für Web-Anwendungen und arbeitet in einem interdisziplinären Team an Usability-Konzepten fürs Web. Dazu streift er täglich stundenlang durchs Web auf der Suche nach Inspiration und entdeckt fortlaufend neue Web-Anwendungen, bei denen muck als *Early Adopter* stets zu den ersten 500 Nutzern zählt. Als bekennender *Webaholic* folgt er jeden Hype und wechselt seine präferierten Social Networks wöchentlich. Die persönlichen Nutzungscharts veröffentlicht er in seinem Weblog. An „guten Tagen“ erreicht muck zu Spitzenzeiten eine Frequenz von 120 Tweets pro Stunde, das API-Limit von 80 HTTP-Requests pro Stunde reicht ihm daher bei Twitter schon lange nicht mehr, weswegen sein Account auf Anfrage auf die „Whitelist für Heavy User“ gesetzt wurde. Ein besonderes Faible besitzt muck für Geo-Dienste. Als *Plazes*[39]-Nutzer der ersten Stunde, stiegt er nach dem Ankauf durch Nokia und der langsamen Verödung des Dienstes (nach einem kurzen Umweg über *Fire Eagle*[40]) zunächst auf *Gowalla*[41] um, landete dann aber bei *Foursquare*[42], das er nun täglich, fast stündlich, manchmal

[39] http://plazes.com/
[40] http://fireeagle.yahoo.net/
[41] http://gowalla.com/
[42] http://foursquare.com/

auch minütlich nutzt, um mit seinem Apple iPhone4 seine aktuelle Geo-Position auf Twitter mitzuteilen.

Michels Persona muck lebt vollständig öffentlich im Web und er sieht sie ein Stück weit als einmalige Kunstfigur, der als Designer außerordentliche Web-Expertise nachgesagt wird und eine unglaubliche Konsequenz und Erfahrung in der Explorierung der Möglichkeiten des Webs. Dies kommt seiner Agentur wirtschaftlich sehr zu Gute und so sieht er es als ständige Aufgabe muck weiterhin im Rampenlicht des Webs zu platzieren.

So offen und intensiv er jedoch das Web nutzt, so würde Michel immer noch behaupten aus seiner Sicht wirklich private Dinge aus dem Web herauszuhalten. muck besitzt nämlich weder Frau noch Kinder, obwohl Michel seit 3 Jahren verheiratet ist und seine Frau schwanger. Auch bemüht er sich, dass im Web keine Privatadresse von ihm auftaucht. Veröffentlichte Geo-Positionen von ihm existieren zwar fast auf dem ganzen Hannoveraner Stadtgebiet, jedoch nicht von seiner Wohnung. Die wirklich private Kommunikation über das Web findet abseits der öffentlichen Social Networks statt. Dort bewegt er sich in viel kleinerem Kreis unter seinem bürgerlichen Namen, achtet hier aber genau darauf, mit wem er auf diese Weise kommuniziert.

Als web-affiner Designer besitzt Michel routinierte Fähigkeiten mit dem Web und seinen Techniken umzugehen, besitzt jedoch kein tiefes Verständnis der Web-Techniken und beschäftigt sich damit auch nicht im wissenschaftlichen Rahmen.

2.3.1.2 Fred Busch

Kurzbeschreibung/Verhaltensmuster Fred beschreibt den Archetypen des Domänenexperten, der sich professionell mit Web-Techniken auseinandersetzt und alle Risiken des Agierens im Web genau einschätzen kann. Er lebt so offen wie aus seiner Sicht für ein zeitgemäßes Auftreten im Web nötig und so privat wie möglich. Er geht außerordentlich bewusst mit persönlichen Daten im Web um und kann die Konsequenzen seines Handelns stets einschätzen.

Fred ist 27 Jahre alt, in Aachen geboren, zur Schule gegangen und hat schließlich auch an der RWTH Aachen Allgemeine Informatik studiert, lebt nun aber in Berlin und arbeitet von dort als freiberuflicher Web-Entwickler für viele europäische Web-Unternehmen. Zwangsläufig setzt er sich dafür mit neuen Web-Techniken auseinander und veröffentlicht zahlreiche selbstgeschriebene Bibliotheken und Skripte unter freien Open-Source-Lizenzen. Er ist Mitglied der W3C HTML Working Group[43]

[43] http://www.w3.org/html/wg/

Name	Michel „muck“ Langhanns
Alter	31 Jahre
Beruf	Web-Designer, -Konzepter (Diplom-Grafikdesigner)
Aktivitäten	Tägliches, hochfrequentes Microblogging (inkl. Multimedia- und Geo-Daten) sowie Social Bookmarking. Eher seltener streamt muck Videos mit seinem iPhone live ins Netz.
Sichtweise auf Problemdomäne	Die Gefahr des Identitätsmissbrauchs sieht er und er ist sich auch dessen bewusst, dass er diesem durch eine Nutzung einer alternativen Architektur vorbeugen kann. Er ist neuem positiv gegenüber eingestellt und fragt eher „Wie kann ich es für mich nutzen?“ statt „Ist es eine Bedrohung für mich?“
Fähigkeiten	Als erfahrener Web-Designer beherrscht er den Umgang mit dem Web sehr gut und kennt die Konzepte hinter vielen Web-Techniken, beschäftigt sich mit diesen allerdings nicht wissenschaftlich.
Interesse/ Motivation	Einerseits ist er als Designer selbst wissbegierig und interessiert sich dafür wie Vertrauenswürdigkeit tatsächlich im digitalen Kontext abgebildet werden kann. Darüber hinaus ist er an einem authentischen Bild seiner Kunstfigur muck im Web interessiert und fürchtet sich vor Identitätsmissbrauch. Die Sicherung der Integrität seiner Inhalte findet er attraktiv.
Ziele (Goals)	Zur Ausübung seines Berufes beschäftigt er sich zwangsläufig intensiv mit dem Web und lebt als Persona vollständig öffentlich. Er profiert dabei geschäftlich von seiner Bekanntheit im Web und er erhält diesen Status der „Omnipräsenz“ durch das ständige Teilen von neuen Inhalten. Die Inhalte selbst sind dabei eher nebensächlich, wichtig ist ihm vielmehr, dass muck an vielen Stellen im Web aktiv erscheint und dabei ein konsistentes, von Michel penibel modelliertes und kontrolliertes Bild geschaffen wird.

Tabelle 2.3: Persona-Steckbrief von Michel „muck“ Langhanns

und dort zur Zeit besonders in die Spezifikation des Canvas-Elements von HTML5 eingebunden. In der Woche verbringt Fred zwei Drittel seines Tages vor seinem MacBook Pro und ist auch in der übrigen Zeit mit dem iPhone in der Tasche online und für seine Kontakte per Mail erreichbar, die er zu großen Teilen am Rande von europäischen Web-Konferenzen kennengelernt hat und nun online täglich über verschiedene Kanäle pflegt. Sein Facebook-Profil gibt nicht außerordentlich viel privates von ihm Preis, sondern nur das, was aus seiner Sicht auch jeder über ihn „zusammengoogelt" könnte, und es dient ihm weniger zur privaten Kommunikation, sondern eher als Visitenkarte und vor allem per *Facebook Connect* als *Single-Sign-On*-Account für das moderne Web. Accounts besitzt er bei unzähligen Social Networks, nutzt davon jedoch nur eine Hand voll aktiv. Dort organisiert er seine Kontakte meist in Gruppen und stellt diesen — sofern möglich — verschiedene Sichtweisen auf seine Inhalte zur Verfügung (je nach tatsächlichem Bekanntheitsgrad). Fred lebt seit mehreren Jahren in einer Beziehung, Informationen über seine Freundin findet man jedoch online kaum und diese können und sollen auch nicht mit ihm in Verbindung gebracht werden.

Fred zeichnet einen außerordentlich bewussten Umgang mit geteilten Inhalten aus. Nicht nur sucht er die Empfänger geteilter Inhalte jedes mal genau aus und wägt anhand des Inhaltes ab, ob die zur Auswahl stehenden Personen diesen wirklich benötigen, wie sie ihn wohl interpretieren werden und ob sie ihn vielleicht gegen ihn verwenden könnten. Aus diesem Grund trifft er auch seine Wortwahl überlegt und doktort an einem 140 Zeichen-Tweet mitunter Minuten herum, bevor er ihn schließlich verwirft, da er den angestrebten Inhalt nicht so in Worte verpackt bekommt, dass dies seinen Ansprüchen genügen würde. Lieber kommentiert er auch einmal zu wenig in Weblogs, als unüberlegt etwas zu veröffentlichen, was man im Anschluss nicht mehr aus dem Web heraus bekommt. Denn dies ist ihm als Informatiker bewusst: Was im Web landet, verschwindet nicht wieder. Um sein Ansehen daher im Web dauerhaft aufrecht zu erhalten, betreibt er Selbstkontrolle und ständige Qualitätssicherung.

Auch wenn Fred Busch kein häufiger Name sein sollte, so existieren dennoch drei verschiedene Fred Buschs, neben ihm recht prominent im Web. Ein Fotograf aus Berlin, ein Amateurfußballer aus Mannheim und ein Musiker, der in ähnlichem Alter wie er ist und eine bei Google prominent verlinkte MySpace-Seite betreibt. Um mit diesen nicht verwechselt zu werden, hat Fred bereits oft *Identitäts-Modellierungs*-Dienste wie

claimID[44] oder *namyz*[45] für sich ausprobiert, findet diese jedoch zu zeitaufwendig in der Pflege, da er nicht jedes mal Lust hat dort einen neuen Account/URL einzutragen, wenn er gerade wieder durchs Web streift und sich testweise bei neuen Anwendungen registriert.

2.3.1.3 Peer Maria Senfmann

Kurzbeschreibung/Verhaltensmuster Peer studierte Kunstgeschichte (M.A.) und beginnt derzeit eine Promotion an der Universität des Saarlandes in Saarbrücken. Als Autodidakt hat er sich neben seiner Arbeit grundlegende Web-Techniken beigebracht und verfolgt mit großen Interesse vor allem die gesellschaftlichen Entwicklungen im Web. Als Skeptiker steht er dabei neuen Anwendungen und Verhaltensweisen zunächst jedoch immer kritisch gegenüber und agiert stets mit Vorsicht und einer Prise Paranoia. „Stellt das Web eine Bedrohung für die Gesellschaft dar?“ — dieser Frage geht er akribisch nach.

Peer ist 31 Jahre alt und hat aufgrund seiner Profession keine besondere Verbindung zum Web, er nutzt es allerdings rege für seine wissenschaftlichen Arbeiten und beschäftigt sich privat mit den gesellschaftlichen Diskussionen, die rund um die Spannungsfelder des World Wide Webs entstehen und wägt Kosten und Nutzen neuer Technologie gegeneinander ab. Ob technischer Fortschritt und die Ausnutzung *aller* Möglichkeiten, die im Web-Kontext entstehen, ein Fluch oder ein Segen ist — er ist sich dessen noch nicht sicher.

Während seines einjährigen Auslandsaufenthaltes in Rom während des Studiums hat Peer viele internationale Freunde gewonnen und mit ihnen Facebook entdeckt, welches sich zu diesem Zeitpunkt noch in der Frühphase befand. Nach seinem Jahr in Italien wurde es zum Hauptkommunikationsmittel der ehemaligen Erasmusstudenten, er kann sich jedoch bis heute nicht dazu durchringen dort unter seinem Klarnamen aufzutreten, sondern benutzt einen fast unaussprechlichen Fantasienamen (ohne Bild). Bekannte von ihm haben so praktisch keine Chance ihn über die Suchfunktion zu finden. Er muss sie ansprechen, wenn sie in Kontakt treten möchten. Denn Peer erfüllt eine besondere Abneigung gegen die wirtschaftliche Nutzung personenbezogener Daten im Web. Es ist nicht so, dass er einen Missbrauch seiner Daten fürchtet und seine Inhalte unbedingt vertraulich behandelt werden

[44] Bei claimID, http://claimid.com/, kann man eigene Inhalte im Web verlinken und damit verifizieren und fremde Inhalte als nicht-eigen kennzeichnen, um Verwechselungen vorzubeugen. Kontakte müssen dann nur noch der Echtheit des claimID-Profils glauben schenken, um dort gesicherte Informationen zu einer Person zu finden.

[45] Die Web-Anwendung Naymz, http://www.naymz.com/, funktioniert auf ähnliche Art und Weise wie claimID.

Name	Fred Busch
Alter	27 Jahre
Beruf	Web-Entwickler, -Konzepter (Diplom-Informatiker)
Aktivitäten	Tägliches „Leben“ im Web. Er probiert mit seinem Facebook-Account als Login viele neue Web-Anwendungen aus. Im Rahmen seiner Arbeit teilt er zahlreiche Bibliotheken und Code-Skripte unter Open-Source-Lizenzen bei *GitHub*[46] und mit seinen Auftraggebern und Kollegen Konzeptdokumente, Grafikvorlagen und sonstige Projektunterlagen über *Dropbox*. Auch Passwörter, Private Keys und Konfigurationen für Web-Server tauscht er über das Web in seinen Entwickler-Teams aus.
Sichtweise auf Problemdomäne	Fred besitzt umfangreiches Fachwissen und kann die Problemdomäne in der Tiefe vollständig durchdringen und für sich bewerten. Er kennt die Problem des Webs hinsichtlich der Schutzziele der IT-Sicherheit und ist sich der Fahrlässigkeit mancher seiner Handlungen im Web bewusst, z.B. wenn er über Facebook vertrauliche Projektdaten austauscht — einfach weil's so bequem ist. Gern würde er seine Web-Kommunikation gegen den unbefugten Zugriff stärker absichern, doch macht er die Erfahrung, dass die dazu aktuell angebotenen Software-Lösungen sich bei der Zusammenarbeit mit Kunden und Kollegen als nicht besonders praxistauglich erweisen und seine Kommunikationspartner den Handlungsbedarf in dieser Richtung auch nicht immer sehen.
Fähigkeiten	Als erfahrener Web-Benutzer beherrscht er den Umgang mit dem Web sehr gut, kennt die Konzepte hinter vielen Web-Techniken und kann diese auch professionell anwenden. In seiner Arbeit an der HTML-Spezifikation setzt er sich auch auf theoretischer und wissenschaftlicher Ebene mit dem Web auseinander.

Tabelle 2.4: Persona-Steckbrief von Fred Busch (Teil 1)

Name	Fred Busch
Interesse/ Motivation	Fred ist sich dessen bewusst, dass seine täglich Kommunikation im Web nicht immer den Ansprüchen genügt, die er sich hinsichtlich des vertraulichen Umgangs mit Daten selbst auferlegt hat oder zu denen er sich teilweise auch per Vertrag verpflichtet hat. Gerne würde er eine vertraulichere Kommunikation betreiben, sieht sich aber im Spannungsfeld zwischen unpraktikablen Software-Lösungen, unwissenden und unfähigen Kunden und schließlich seinem Wissen über diesen Missstand im Web, das er nicht richtig ausgelöst kriegt.
Ziele (Goals)	Das Teilen im Web macht Fred auch Freude. Das Teilen in schön umgesetzten Social Networks noch mehr. Eines seiner Ziele, das ihn auch daran hindert komplexere Software auszuprobieren, ist der Spaß, den er im Web durch das Teilen von Inhalten haben kann. Er lebt das Web mit Leidenschaft, ist begeistert von den vielen neuen Möglichkeiten die es erschafft und immer versucht sie auszuprobieren, wäre da nicht die Selbstregulation, die es ihm teilweise verbietet. Sich frei im Web bewegen zu können, ohne Konsequenzen für sein Ansehen, seine Reputation und seine Arbeit zu fürchten, ist ein weiteres Ziel von Fred.

Tabelle 2.5: Persona-Steckbrief von Fred Busch (Teil 2)

müssten (so brisant schätzt selbst er sie nicht ein), vielmehr gönnt er es den Betreibern der Web-Anwendungen nicht, interne Profile ihrer Nutzer zu erstellen und aus dieser Wissensbasis (womöglich auch erst indirekt) Profit zu schlagen. Dies ist keine Währung, in der er für die Dienstleistung des Unternehmens (Entwicklung und Bereitstellung der Web-Anwendung) bezahlen möchte, obgleich er sich dessen bewusst ist, dass Unternehmen Web-Anwendungen nicht kostenlos zur Verfügung stellen können.

Seine Bemühungen richten sich dabei nicht zwangsläufig gegen alle Unternehmen im Web, sondern vor allem gegen die „Großen des Webs“: Google, Facebook, Amazon, Yahoo!, ebay/PayPal, mittlerweile auch Apple. Der Gedanke, dass einer von diesen zu übermächtig wird, bereitet ihm Bauchschmerzen — ohne, dass er eine konkrete Vorstellung davon hätte, was dies für die Gesellschaft in der er lebt, letztendlich bedeuten würde.

Nichtsdestotrotz möchte Peer das Web und seine Möglichkeiten nutzen und hantiert dabei immer ein bisschen aufwendiger als der Durchschnittsnutzer: Anti-Spionage- und Anti-Tracking-Software laufen dauerhaft und sein Web-Browser ist auf ein minimales Featureset (kein Java, kein JavaScript, keine Cookies, PopUp- und Werbeblocker) beschränkt, damit Webseiten möglichst keine versteckten Aktionen ausführen, Cookies speichern oder auslesen oder gar seine Interaktion aufzeichnen können. Bei neuen Web-Anwendungen registriert sich Peer, wenn überhaupt nur unter Angabe falscher Personendaten und mit Wegwerf-E-Mail-Adressen. Es existiert wohl kein Social Network dessen intendiertes Nutzungsszenario er damit direkt erfüllt. So bleiben ihm viele Funktionen meist vorenthalten, da sie von ihm mehr Offenheit gegenüber der Anwendung und gegenüber des Webs erfordern würden.

Fred ist sich der durch sein Verhalten ergebenen Bedienungsschwierigkeiten bewusst und nimmt es in Kauf, wenn Web-Anwendungen zunächst nicht funktionieren oder Darstellungsfehler auftreten, bis er ihnen die notwendigen Rechte erteilt hat, bestimmte Skripte ausführen zu dürfen. Eine Alternative zu seinen Schutzvorkehrungen sieht er aktuell jedoch nicht, auch wenn er es selbst als Dilemma bezeichnet: Das Web könnte so wunderbare Möglichkeiten der Kommunikation bieten, doch sind sie aufgrund der aktuellen Implementierung für ihn unbenutzbar.

2.3.1.4 Maximilian Prange

Kurzbeschreibung/Verhaltensmuster Maximilian „Max“ Prange steht für den unbedarften Nutzertyp, der mit dem Web aufgewachsen ist und in erster Linie seine Vorzüge sieht. Mit seinen Freunden teilte er über Jahre zahlreiche Inhalte in

Name	Peer Maria Senfmann
Alter	31 Jahre
Beruf	Wissenschaftlicher Mitarbeiter und Promotionsstudent an der Universität des Saarlandes (Kunstgeschichte, Master of Arts)
Aktivitäten	Tägliche Recherche wissenschaftlicher Artikel im Web, häufige Teilnahme an Diskussionen in Fachforen. „Kontaktpflege“ zu ehemaligen Erasmusstudenten über Facebook.
Sichtweise auf Problemdomäne	Peer sieht die Möglichkeiten, die das Web bietet, diffus vor sich, kann deren Tragweite allerdings noch nicht richtig abschätzen. Er geht daher sehr skeptisch und vorsichtig mit persönlichen Inhalten im Web um und hinterfragt bei neuen Anwendungen zunächst: Sind sie eine Bedrohung für meine Person?
Fähigkeiten	Peer ist geübter Anwender des Webs. Zwar besitzt er kein Fachwissen, hats ich allerdings ambitioniert technische Grundlagen erarbeitet und verfügt eher über ein überdurchschnittliches Verständnis seiner Browser-Software und deren Extensions zum Verschleiern und Verbergen seiner Online-Aktivitäten. Manuelle Konfigurationen von lokalen Proxy-Servern, die Werbe-Inhalte und Tracking-Codes aus Webseiten herausfiltern, gelingen ihm letztlich, auch wenn er sich dafür durch unzählige Foreinträge wühlen muss.
Interesse/ Motivation	Gerne möchte Peer vertraulicher und abseites der zentralen Social Networks Inhalte teilen, nicht aus Sorge um Missbrauch seiner Inhalte, sondern, damit kein Unternehmen übermächtiges Wissen über zu viele Akteure des Webs erlangt. Die Konsequenzen dessen fürchtet er, auch wenn er sie sich noch nicht ausmalen kann.
Ziele (Goals)	Der Kontakt zu seinen ehemaligen Erasmusstudenten ist Peer sehr wichtig. Auch im Rahmen seiner Promotion schätzt er deren Feedback zu eigenen Ideen und die sich oft ergebenen Fachdiskussionen. Zwar bietet die Universität bereits Web-Plattformen für einen derartigen Austausch an, doch scheint Facebook für den alltäglichen Austausch in seiner Peer-Group nach wie vor unersetzlich. Es dient z.B. auch als Plattform, um neue Kontaktdaten auszutauschen. Insofern muss er zähneknirschend die Nutzung in Kauf nehmen, würde allerdings gerne ein alternatives Social Network vorschlagen.

Tabelle 2.6: Persona-Steckbrief von Peer Maria Senfmann

altersgerechten Social Networks, wie SchülerVZ oder Facebook, und machte sich wenig Gedanken über seine Reputation und die Wirkung seiner Inhalte auf andere Personen als seine Freunde. Während einer Projektwoche seiner Schule zum Thema „Privatsphäre im Web“ ist er kürzlich erstmals mit dem Thema konfrontiert worden und entdeckt im Moment den bewussteren Umgang mit Inhalten im Web.

Max ist gerade 18 Jahre alt geworden und bereitet sich auf das Abitur vor. Vor zwei oder drei Jahren hat ihn sein bester Freund ins SchülerVZ eingeladen und wenig später war die halbe Schule dort versammelt. Er genoss es sehr sich „im VZ“ über den Unterricht, die Lehrer und die Schule im allgemeinen auszulassen und teilte dort auch Unterrichtsmaterialien, Links zu Sekundärlektüren aktueller Literatur aus dem Deutschunterricht und Übersetzungen der Texte ihres Lateinbuches. Seitdem er sein erstes Fotohandy besitzt entstanden zusätzlich unzählige Fotos von Partys und aus seinen Urlauben. Das „VZ“ wurde nach und nach zum primären Kommunikationsmittel im Freundeskreis und die internen Nachrichten dort ersetzten mit der Entstehung mobiler Clients für die Web-Anwendung auch zunehmend die teure SMS-Nachricht. Max könne gar nicht genau beziffern, wie viele Inhalte er wohl je dort geteilt hat und benennen wer darauf aktuell welchen Zugriff besitzt. Denn letzteres war ihm bislang soweit unwichtig, so lange seine engen Freunde allen Inhalte sehen konnten.

Seit der Projektwoche seiner Schule zum Thema „Privatsphäre im Web“ macht er sich erstmals Gedanken über Konsequenzen seines Handelns im Web und ob unter seinen über 200 SchülerVZ-Kontakten, die er lang nicht alle persönlich kennt und die auch nicht alles Schüler seiner Schule sind, etwa „schwarze Schafe“ stecken, die seine Inhalte vielleicht eher nicht zu Gesicht bekommen sollten. Da er allerdings bisher viel Spaß im Web hatte, auch mit ihm unbekannten Personen, ist er sich unsicher, ob er dies wirklich aufgeben sollte, um einer Gefahr vorzubeugen, die er nicht mal bennen kann.

2.3.1.5 Heinrich Schulte-Hofland

Kurzbeschreibung/Verhaltensmuster Heinrich verkörpert den technikfernen Web-Anfänger im mittleren Alter (60+), der zwar als Frührentner genug Zeit hat, sich mit dem Web auseinanderzusetzen, doch sich dabei nicht eigenständig neue Dinge beibringt. Er ist meist auf die Hilfe seiner Kinder angewiesen, die ihm den Umgang mit dem Web erläutern und ihm seinen Computer und Accounts in Web-Anwendungen entsprechend einrichten. Das Web fasziniert ihn trotzdem, weil ihm dadurch die Möglichkeit gegeben wird, mit seinen im Ausland lebenden Verwandten in Kontakt zu bleiben.

Name	Maximilian Prange
Alter	18 Jahre
Beruf	Schüler
Aktivitäten	Tägliche Nutzung des Webs zur Kommunikation mit Freunden. Bevorzugte Social Networks: SchülerVZ und Facebook. Geteilt werden vor allem private Nachrichten und Links auf andere Web-Inhalte sowie eigene Fotos und Videos von Partys und aus Urlauben.
Sichtweise auf Problemdomäne	Max ist mit dem Web aufgewachsen und nutzte es bisher selbstverständlich und sorglos. Seine Eltern lehrten ihn keine besondere Vorsicht, erst durch eine Projektwoche seiner Schule wurde er auf die Gefahren des allzu öffentlichen Teilens im Web aufmerksam. Seitdem beschäftigt er sich mit diesem Thema gedanklich, ist sich jedoch der persönlichen Bedeutung für ihn noch unsicher.
Fähigkeiten	Max ist mit Computern und dem Web aufgewachsen. Schon früh in seiner Kindheit hatte er Zugriff auf einen PC und besitzt seit seinem 15 Lebensjahr einen eigenen Rechner mit Internetzuganng in seinem Zimmer. Als durchschnittlich Technik-interessierter kann er dennoch die Grundkomponenten eines PCs benennen und hat auch schon einmal seine Grafikkarte selbst ausgetauscht. Das Web kennt er ausschließlich aus Flatrate- und Breitband-Zeiten und nutzt es täglich völlig selbstverständlich. Die Techniken des Webs sind ihm in Grundzügen geläufig, aber weder hat er sich je mit Protokollspezifikationen beschäftigt noch mit Markup-Sprachen oder Datenpersistenz. Er nimmt vollkommen die Rolle des geübten Anwenders ein.
Interesse/ Motivation	Max ist sensibilisiert für das Thema „Privatsphäre im Web“ und möchte gerne mehr darüber erfahren, um beurteilen zu können, ob es für ihn relevant ist. Im Rahmen der Projektwoche hat der die Konzepte Vertraulichkeit, Verfügbarkeit und Integrität kennengelernt, hält jedoch eine Gefahr für ihn persönlich für abstrakt und etwas weit hergeholt.
Ziele (Goals)	Aus Max’ Sicht gehört das Teilen von Inhalten im Web zu einer normalen Kommunikation unter Freunden. Das Teilen innerhalb von SchülerVZ bereitet ihm viel Spaß und gerade der mobile Zugriff erleichtert auch das Treffen von Verabredungen oder das Organisieren eines immer verfügbaren Adressbuchs. Max möchte gerne weiterhin das Web für seine tägliche Kommunikation nutzen, einfach weil er es für selbstverständlich hält.

Tabelle 2.7: Persona-Steckbrief von Maximilian Prange

Heinrich würde sich selbst nie als Technik-affin bezeichnen, doch sei er auch nicht auf den Kopf gefallen. Über 30 Jahre lang führte er einen Buchladen in Kleve bis er vor kurzem in Ruhestand ging und das Geschäft seinen Töchtern überließ. Seitdem entdeckt er Dank eines geschenkten MacBooks das Web jenseits der Buchkataloge neu und schätzt dabei besonders die Möglichkeiten mit seiner in Athen lebenden Schwester und seinem ebenfalls dort lebenden Patenkind in Kontakt zu bleiben. Oft schaut er Fotos in deren *Flickr*-Account an und schreibt kurze Kommentare dazu. Seit kurzem besitzt seine Schwester eine neue Kamera, die nun auch direkt Videos zu YouTube hochladen kann. Da er jedoch keine RSS-Feeds abonniert hat, informiert ihn seine Schwester per E-Mail über jedes Update. Ein bis zweimal pro Woche schaut er diese durch und wartet dann bis seine Frau nach Hause kommt, bevor sie sich zusammen die neuen Fotos oder Videos anschauen. Zu Weihnachten schenken ihm seine Töchter eine Videokamera. Dann soll er lernen wie auch er Videos ins Web stellen kann.

Das Web sieht er als offensichtlich wichtige Errungenschaft der Neuzeit, da seine Töchter ihm davon nur positives berichren. Allerdings zählt er sich dadurch eigentlich auch schon nicht mehr zur originären Zielgruppe und glaubt es sei eher etwas für jüngere Leute. Ein tiefes Verständnis aller mit dem Web verbundenen Techniken und Möglichkeiten beansprucht er daher nicht, ihm genügt es, wenn seine Töchter ihm zeigen wie er damit umgeht und wie genau er einzelne Inhalte teilen kann. Die ihm bekannten Wege verlässt er dann möglichst nicht mehr, weil er sich in ihm unbekannten Web-Terrain aufgrund seiner technischen Unwissenheit nur sehr unsicher forbewegen kann. Der elektronsiche Austausch von Videos und Fotos innerhalb der Familie gefällt ihm jedoch sehr und er ist neugieirig darauf bald eigene Videos für das Web produzieren zu können.

Dass sich außer seiner Familie noch unzählige weitere Personen im Web herumtreiben und dabei möglicherweise auch hochgeladene Videos von ihm einsehen können, wenn er einmal die Privatsphäre-Einstellungen versehentlich falsch gesetzt hat, dies ist ihm nicht so richtig bewusst, da ihm Web-Mashups aus öffentlichen Inhalten völlig unbekannt sind und er bei weitem nicht Abschätzen kann wie schnell sich Inhalte im Web verbreiten können, wenn diese öffentlich zugänglich sind. Würde man ihn darauf ansprechen und ihm alternative Möglichkeiten des Teilens aufzeigen, so wäre er dann sicherlich daran interessiert dies zu lernen.

2.3.2 Kontext- und Aktivitätsmodelle

Ergänzend zu den Benutzermodellierungen sollen auch relevante Aktivitäten der Benutzer und deren Kontexte in Modelle überführt werden. Hierbei erscheinen

Name	Heinrich Schulte-Hofland
Alter	63 Jahre
Beruf	Rentner
Aktivitäten	Gelegentliches Schreiben von E-Mails an seine Familienmitglieder, regelmäßiges Ansehen von im Web gespeicherten Videos und Fotos seiner Schwester in Athen.
Sichtweise auf Problemdomäne	Heinrich ist sich des in dieser Ausarbeitung beschriebenen Problems kaum bewusst und hätte auch Schwierigkeiten es in einer Fachdiskussion nachzuvollziehen, da ihm das Web an sich völlig fremd ist und er lediglich den Umgang mit einzelnen Web-Anwendungen erlernt hat. Aufgrund seiner Persönlichkeit kann jedoch angenommen werden, dass er sich für einen vertraulichen Umgang mit eigenen Inhalten im Web interessiert, wenn er ähnlich einfach zu praktizieren wäre.
Fähigkeiten	Heinrich besitzt zwar moderne Hardware, die er geschenkt bekommen hat, allerdings kein technisches Fachwissen — weder von Computern im Allgemeinen, noch vom Web im Speziellen. Alle Online-Aktivitäten, die er ausübt, hat er durch seine Familie beigebracht bekommen und kann sie nach einiger Übung auch alleine durchführen. Ändert sich jedoch die Benutzungsoberfläche oder treten Fehler durch versehentliche Eingaben auf, hat er Schwierigkeiten mit der Bedienung oder mit der Ursachenklärung und -lösung.
Interesse/ Motivation	Das Problem, dass auch private Inhalte von Heinrich öffentlich werden könnten (z.B. durch versehentlich getätigte Einstellungen), ist ihm nicht richtig bewusst. Dies zu verhindern wäre jedoch in seinem Interesse, würde ihn darüber aufklären.
Ziele (Goals)	Heinrich nutzt das Web in erster Linie, um mit seiner Familie in Kontakt zu bleiben. Er erkennt, dass dies heutzutage ein aktuelles Kommunikationsmittel ist und akzeptiert, dass er sich damit auseinander setzen muss. Da er hierzu allerdings initial meist auf die Anleitung seiner Töchter angewiesen ist, ist er bemüht die bereits erlernten Fähigkeiten zu verinnerlichen und macht sich so gut es geht handschriftliche Aufzeichnungen dazu, um selbstständig zu bleiben und nicht immer seine Töchter um Mithilfe bitten zu müssen. Das Teilen von Inhaltem im Web erfüllt ihn daher mit Freude, solange er sich in einem unveränderten, ihm vertrauten Kontext bewegen kann.

Tabelle 2.8: Persona-Steckbrief von Heinrich Schulte-Hofland

für den weiteren Prozess vor allem die beiden zentralen Prozessketten „Das Teilen von Inhalten“ sowie „Die Entstehung von Kontakten“ wichtig. Denn ersteres sollte die neuartige Architektur unmittelbar ermöglichen und damit dies unter Berücksichtigung der Schutzziele stattfinden kann, ist eine gesicherte Identität der Online-Kontakte notwendig.

2.3.2.1 Das Teilen von Inhalten

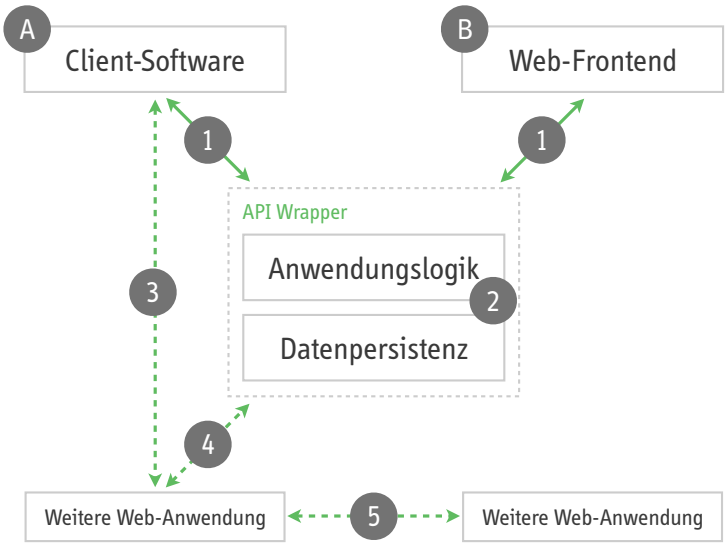

Abbildung 2.7: Kontext des Teilens. Das Gefüge aus Anwendungen im Web.

Abbildung 2.7 stellt den Kontext des Teilens dar. Ein Inhalt entsteht entweder durch Nutzung einer Client-Software einer Web-Anwendung (A; besonders von der Anwendung Twitter bekannt) oder durch Nutzung der Möglichkeiten des Web-Frontends einer Anwendung (B). Von dort gelangt er in jedem Fall über die öffentliche Schnittstelle der Anwendung (API) zur Anwendungslogik (1) und von dort schließlich zur Datenpersistenz (2). Der Inhalt befindet sich nun innerhalb der Web-Anwendung und kann von dort über den gleichen Weg zurück auch wieder an Empfänger ausgeliefert werden. Inhalte können vor allem bei der Nutzung von Client-Software auch in mehreren Anwendungen gleichzeitig geteilt werden (3) oder direkt zwischen Web-Anwendungen im Rahmen eines Mashups über deren Programmierschnittstellen (4). Auch dritte Anwendungen, die im Moment des Teilens noch kein beabsichtigter Zielort des Inhaltes sind, können zu späteren Zeitpunkten über die Schnittstellen zwischen Anwendungen Inhalte erhalten (5).

Abbildung 2.8 stellt den sequentiellen Ablauf innerhalb des Kontextes aus Abbildung

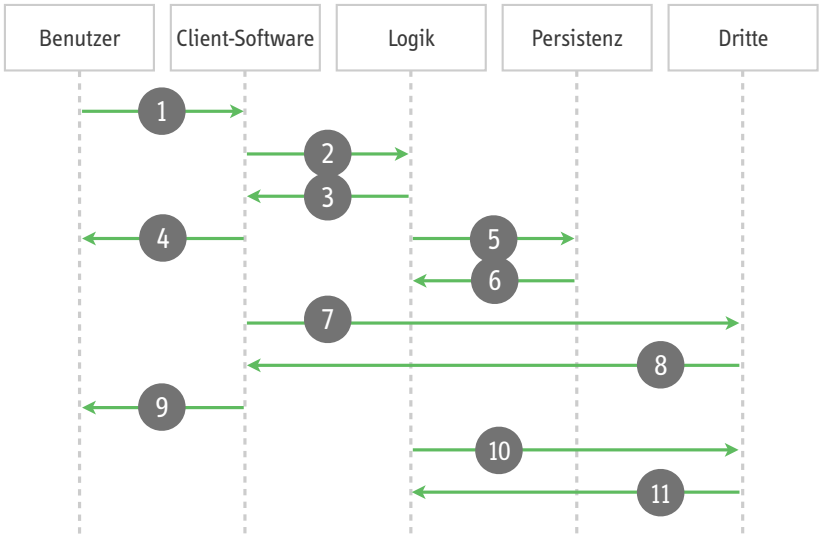

Abbildung 2.8: Ablauf des Teilens. Handlungssequenzen von Benutzer und Systemen.

2.7 dar. Ein Benutzer interagiert solange mit einer Client-Anwendung (kann auch ein Web-Frontend als Client sein) bis ein Inhalt schließlich veröffentlicht werden soll (1). Die Client-Anwendung übermittelt diesen an die Anwendungslogik (2) und diese quittiert den Erhalt (3), was wiederum an den Benutzer als Feedback weitergereicht wird (4). In den meisten Fällen werden Inhalte dann innerhalb der Anwendung persistiert (5) mit einem entsprechendem Feedback des Persistenz-Layers (6). Dies kann auch schon *vor* der positiven Rückmeldung der Logik an den Client (Schritt 3) erfolgen und diese kann auch davon abhängig gemacht werden. Client-Anwendungen können Inhalte anschließend auch dritten Web-Anwendungen zur Veröffentlichung zur Verfügung stellen können (7) und entsprechendes Feedback (8) über Erfolg oder Misserfolg an den Benutzer weiterreichen (9). Schließlich sei hier auch der Fall berücksichtigt, dass Anwendungen selbstständig Inhalte ihrer Benutzer an Dritte weiterreichen können (10), in den meisten Fällen erneut mit Feedback zur Aktion (11).

2.3.2.2 Die Entstehung von Kontakten

In Abbildung 2.9 ist die sequentielle Abfolge der Aktionen aufgetragen, die zum „Befreunden“ zwischen Benutzern in Web-Anwendungen führt. Das sehr einfache Modell ist weniger trivial als es erscheint. Denn zu unterscheiden sind im wesentlichen drei verschiedene Arten der Interaktion:

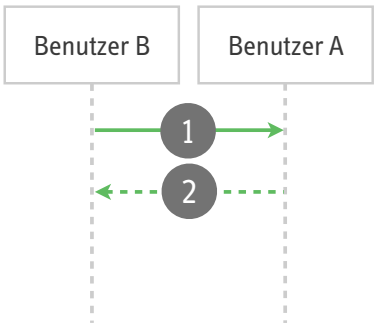

Abbildung 2.9: Die „Befreundung“ von Benutzern in Web-Anwendungen ist weniger trivial als sie in dieser sequentiellen Form erscheint.

- das durch Twitter populär gewordene „Folgen“, welches bei Facebook als *Fan*-Sein umgesetzt wird,
- die klassische *Zwei-Wege-Handschlag*-Freundschaft
- die Freundschaftsanfrage mit Ignorier-Möglichkeit

Beim „Folgen“ entscheidet sich Benutzer A in den meisten Fällen dafür Inhalte von Benutzer B zu abonnieren. Im Moment des Klicks (1) stehen ihm diese unmittelbar zur Verfügung. Benutzer B wird darüber informiert und kann, wenn überhaupt, erst nachträglich Benutzer A verbieten seine Inhalte einzusehen (2).

Der Zwei-Wege-Handschlag ist typisch für die meisten Web-Anwendung: Benutzer A findet eine öffentliche Repräsentation eines Benutzerprofils von Benutzer B und stellt eine Freundschaftsanfrage (1). Bis zur Entscheidung von B bleiben die Inhalte B für A zunächst noch verborgen. Bestätigt B die Freundschaft (2) so hat dieser daraufhin Zugriff auf die Inhalte von A und vice versa. Entscheidet er sich gegen die Freundschaft (2), wird A darüber informiert und kann dann ggfs. eine neue Anfrage stellen.

Eine Abwandlung davon findet man immer häufiger im Web, um Benutzer nicht in die Verlegenheit zu bringen Anfragen aus Höflichkeit annehmen zu müssen: Stellt ein Benutzer A eine Anfrage an B (1), so wird B darüber informiert, hat aber neben der Annahme auch die Möglichkeit die Anfrage schlicht zu ignorieren. Für B wird sie dann ausgeblendet, A erhält darüber jedoch kein Feedback und ihm bleibt unklar, ob die Anfrage nur übersehen oder absichtlich ignoriert wurde.

2.3.3 Inhaltsmodell

Für den weiteren Entwurf ist es außerdem hilfreich ein Modell zu besitzen, welches die potentiell teilbaren Inhalte und ihre Eigenschaften beschreibt. Als wesentliche Charakteristika eines nutzergenerierten Inhaltes sind festzuhalten:

Benutzerzugehörigkeit Ein Inhalt besitzt einen Urheber. Dieser entscheidet über die Ausgestaltung und alle Verwendungen des Inhaltes. In wenigen Fällen geht die Urheberschaft auf mehrere Personen über, die einen Inhalt gemeinsam erstellt haben[47].

Payload Ein Inhalt besitzt immer eine Kerninformation, diese wird umgangssprachlich als der eigentliche Inhalt angesehen. Die Ausgestaltung kann verschieden sein: Es kann Text oder ein Verweis im Web sein oder Binärdaten (Fotos, Videos, binäre Dokumente) oder sogar ein Datenstrom.

Empfänger/Autorisierte Teilhaber Ein Inhalt besitzt eine Gruppe durch den Urheber bestimmter Personen/Accounts, denen der Inhalt primär zur Verfügung gestellt werden soll. Diese Aufgabe hat in der Regel die Anwendungslogik einer Web-Anwendung durchzuführen.

Metadaten Neben Informationen zu Urheber, Empfänger und dem eigentlichen Payload, können Inhalte eine Reihe von Metadaten besitzen. Dies können inhaltstypabhängige Daten sein, z.B. EXIF-Daten bei Fotografien im JPEG-Format oder Referenzierungen anderer Inhalte im Web (URL auf Inhaltsobjekt im Falle eines Kommentars darauf, z.B. Antwort auf Tweet bei Twitter), oder allgemein im Web nutzbare Metadaten, z.B. zur Nutzungslizenz (Steht dieser Inhalt unter dem Copyright oder unter einer freien Lizenz?) oder zur Geo-Position des Urhebers zum Zeitpunkt der Inhaltserstellung.

2.3.4 Modellierung von Vertrauenswürdigkeit

Als weiteres relevantes Modell sollen schließlich die drei erkannten Parameter zur Beeinflussung von wahrgenommener Vertrauenswürdigkeit zueinander in Beziehung gesetzt werden:

[47] Dieser Punkt wird zu einem späteren Zeitpunkt separat betrachtet

- die Ausprägung von Transparenz und Selbstbeschreibungsfähigkeit einer Web-Anwendung
- die Ausprägung der Bewertungsmöglichkeiten der Anwendung durch externe Autoritäten
- die Ausprägung der Dokumentationsfähigkeit der eigenen Erfahrungen mit der Web-Anwendung

Abbildung 2.10 stellt sie in einem dreidimensionalen System dar. Nur wenn sich die Ausprägungen aller Faktoren im grünen Bereich befindet, kann von einer ausreichend modellierten Vertrauenswürdigkeit der eigenen Anwendung ausgegangen werden.

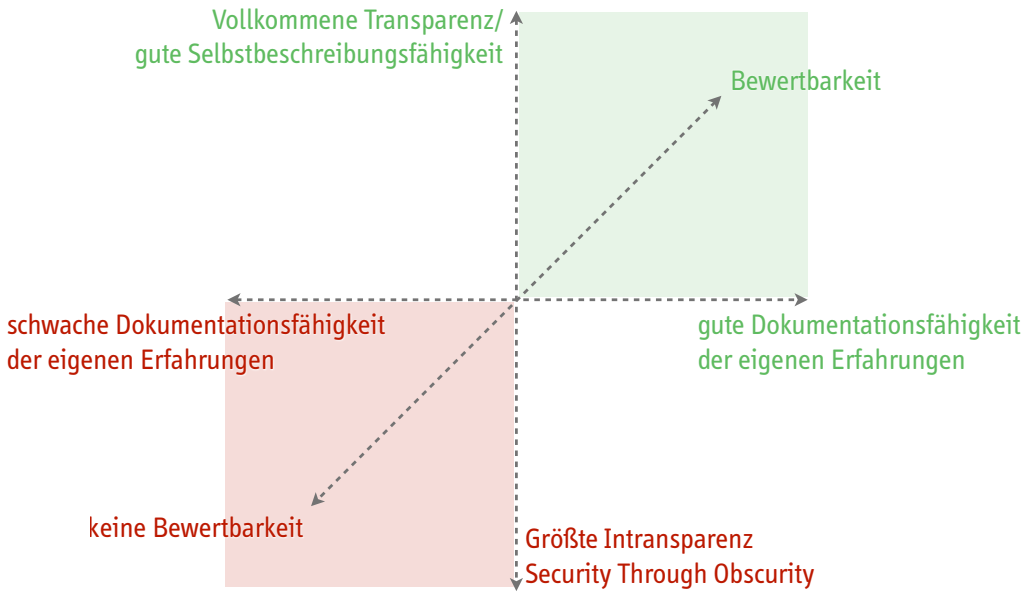

Abbildung 2.10: Parameter zur Beeinflussung wahrgenommener Vertrauenswürdigkeit. Grün kennzeichnet eine angemessene Erfüllung, rot eine nicht ausreichende Erfüllung zur Etablierung von Vertrauenswürdigkeit.

2.4 Requirements Definition

Im Anschluss an die verschiedenen Modellierungen relvanter, bisheriger Erkenntnisse im Rahmen der *Modeling*-Phase sollen nun die tatsächlichen Anforderungen an die zu entwerfende Architektur ermittelt und dokumentiert werden. Es geht in diesem Schritt um die Beantwortung der Frage „*Was* soll die neue Architektur leisten?“ bevor danach in der *Design Framework*-Phase erstmals auf das *Wie?* eingegangen wird. Eine zentrale Methode im Rahmen der Anforderungsermittlung werden Persona-basierte Kontextszenarien sein, die eine ideale Interaktion einer Persona mit der neuen Architektur in einem bestimmten Nutzungskontext narrativ beschreiben. Diese Herangehensweise haben die Autoren des Goal-directed Designs an das *Scenario-based Design* von Rosson und Carroll (2001) angelehnt, stellen dabei jedoch die Rolle der Persona als Akteur als wichtiges Element in den Vordergrund, während Rosson und Carroll eine abstrakte Nutzerrolle als Akteure der Szenarien empfehlen [RC01].

Um über die Formulierung von Kontextszenarien schließlich auf Anforderungen zu schließen, sind nach dem Goal-directed Design die folgenden fünf Aktivitäten durchzuführen:

1. Creating problem and vision statement
2. Brainstorming
3. Identifying persona expectations
4. Construction context scenarios
5. Indentifying requirements

Um Designern und Entwicklern zumindest eine grobe Arbeitsrichtung bei der Ausarbeitung der bedeutenden Kontextszenarien vorzugeben, soll mit einer **Formulierung des aktuellen Kernproblems und einer Vision dessen Lösung** gestartet werden. Problembeschreibungen wurden in dieser Ausarbeitung bereits mehrfach an verschiedenen Stellen formuliert, dennoch soll im Abschnitt 2.4.1 eine komprimierte Problemformulierung der Vollständigkeit halber und als wichtige Wiederholung vor der Anforderungsermittlung noch einmal angeführt werden.

Das anschließende **Brainstorming** dient bei der Arbeit in Teams vor dem Beginn der Szenarienerstellung vor allem dazu den Geist frei von möglicherweise schon umherschwirrenden, zu konkreten Lösungsvorschlägen zu bekommen. Indem frei in der Gruppe über ein Thema nachgedacht wird, werden so viele (auch absurd erscheindende) Ideen in den Raum geworfen, dass eine Person anschließend eher dazu

in der Lage sein soll, ihre eigenen Ideen loszulassen und erneut in neue Richtungen zu denken. Diese Brainstorming-Phase ist als Aktivität in der Ausarbeitung nicht weiter dokumentiert.

Bei der **Identifizierung der Erwartungen** verschiedener Personas sei auf die detaillierte, narrative Beschreibung der Personas in Abschnitt 2.3.1 verwiesen. An dieser Stelle wurden die Personas bereits facettenreich ausgeschmückt und ihre Fähigkeiten, Ansichten, Sehnsüchte und eben auch Erwartungen an eine neue Architektur beschrieben. Diese dienen an dieser Stelle erneut als *Input*.

Schließlich werden im vierten Schritt die eigentlichen **Kontextszenarien** formuliert. Das Goal-directed Design legt hier nahe, darin die folgenden Fragen zu klären:

- In welchem Kontext wird das Produkt eingesetzt?
- Wie lange wird es jeweils genutzt?
- Wird die Persona häufig bei der Benutzung unterbrochen?
- Arbeiten mehrere Personas an einem Computer oder Gerät?
- Mit welchen anderen Produkten wird das neue Produkt zusammen genutzt?
- Welche primären Aufgaben müssen Personas erfüllen, um ihre Ziele zu erreichen?
- Was ist das erwartete Endergebnis nach der erfolgreichen Nutzung des Produktes?
- Wieviel Komplexität ist zulässig, abhängig von den Fähigkeiten der Persona und der Häufigkeit der Nutzung?

Bei der Betrachtung der Fragen wird erneut deutlich, dass dieses Vorgehensmodell besonders eine visuelle Gestaltungslösung als Endprodukt des Entwicklungsprozesses anstrebt und als Anwendungskontext eine wirtschaftlich-motivierte Arbeitsumgebung annimmt. Es wird vor allem auf die Interaktion mit der Architektur Wert gelegt und auf die Qualitäten eines „Produktes“, welches sich mitunter auch vermarkten lässt. In diesem Falle jedoch steht die endgültige visuelle Gestaltung einer Benutzungsoberfläche nicht im Mittelpunkt der Entwurfsaktivitäten, sondern der Entwurf und die Kommunikation eines Systemdesigns. Die vorgeschlagene Herangehensweise an die Kontextszenarien wird daher im Rahmen dieser Ausarbeitung an die veränderten Gegebenheiten und Ziele angepasst, folgt aber weiterhin dem Ziel, die Ermittelung von Anforderungen zu ermöglichen. Abschnitt 2.4.2 dokumentiert auszugsweise mehrere im Rahmen der Anforderungsermittlung erstellte Kontexts-

zenarien, die für den Systementwuf in Abschnitt 2.5 von besonderer Bedeutung sind.

Nach der Formulierung der Kontextszenarien werden diese hinsichtlich ihrer *Aktionen*, *Objekte* und *Kontexte* analysiert und daraus **Anforderungen abgeleitet**. Durchgeführte Aktionen referenzieren unmittelbar Funktionalitäten der neuen Architektur, Objekte verbergen Hinweise auf datenbezogene Anforderungen und erläuterte Kontexte können ebenfalls Anforderungen an die Nutzungskontexte der Architektur beschreiben.

2.4.1 Problem und Vision

Das *Problem statement* soll auf abstrakter Ebene noch einmal den Zweck der Entwurfsaktivitäten verdeutlichen, und damit als Inspiration für die Kontextszenarien dienen. Es soll eine Situation beschreiben, die einer Veränderung bedarf, und es stützt sich dabei unmittelbar auf die Erkenntnisse aus der Recherche und der Modellierungen:

Menschen können eigene Inhalte im Web nicht frei teilen, um damit uneingeschränkt ihre eigentlichen, individuellen Ziele zu verfolgen, sondern sie müssen die Auswahl der geteilten Inhalte einschränken, ihr Verhalten selbstregulieren und Identitätsmissbrauch und Datenverlust fürchten, weil ihnen beim Teilen eigener Inhalte im Web keine ausreichende Vertraulichkeit, Integrität und Verfügbarkeit gewährleistet werden kann.

Das *Vision statement* lautet als Umkehrung des *Problem statements*:

Eine neue Architektur zum Teilen nutzergenerierter Inhalte im Web ermöglicht Menschen das freie, uneingeschränkte Teilen eigener Inhalte im Web und dadurch die Erreichung ihrer individuellen Ziele, in dem sie Vertraulichkeit, Integrität und Verfügbarkeit der eigenen Inhalte gewährleistet und dies auch für die Menschen nachvollziehbar und vertrauenswürdig gestaltet.

2.4.2 Kontextszenarien

Im Folgenden werden im Rahmen dieser Ausarbeitung vier der erstellten Kontextszenarien exemplarisch für die Fülle an in der Anforderungsermittlung benutzten Szenarien dokumentiert. Diese hier aufgeführten Szenarien besitzen Schlüsselfunktionen bei der weiteren Gestaltung der Architektur.

2.4.2.1 Fred Busch wünscht Feedback

Um von seinen Arbeitskollegen Feedback zur Umsetzbarkeit spezifischer Kundenwünsche bei der Entwicklung einer neuen Software zu erhalten, möchte Fred Busch seinen Kollegen ein vom Kunden per E-Mail zugeschicktes Konzeptdokument zur Verfügung zu stellen.

Fred befindet sich in seinem Büro in Berlin-Mitte, dass er sich mit vier weiteren Freiberuflern teilt, die zur Zeit essen sind. Er sitzt an seinem Schreibtisch, auf dem sich außer ein paar DIN A3-Blättern mit selbstgezeichneten Skizzen, Stiften, seinem Mobiltelefon, einer Bluetooth-Tastatur und -Maus, nur noch sein 27"-Display befindet, welches an sein MacBook Pro angeschlossen ist. Eben hat er eine neue E-Mail von seinem aktuellen Kunden erhalten und blättert das angehängte Konzeptpapier in seinem PDF-Viewer durch.

Zu spezifischen Punkten des Konzeptes möchte Fred gerne Feedback seiner Arbeitskollegen erhalten, die mit ihm zusammen die Software umsetzen werden. Er markiert dazu einzelne Absätze des Dokumentes und versieht sie mit kurzen Kommentaren. Anschließend zieht er die gespeicherte Datei auf ein Programmsymbol im Dock von Mac OS X (Symbol einer Client-Software zum vertraulichen Teilen des Dokumentes) und wählt aus seinem Adressbuch die vier Kontakte seiner Arbeitskollegen aus. Da das Dokument vertraulich ist, legt er fest, dass es von ihnen nicht weitergegeben werden darf. Da er weiterhin dem Kunden zügig antworten möchte, möchte Fred sicher gehen möchte, dass das Dokument seine Kollegen innerhalb der nächsten zwei Stunden erreicht und sie darauf reagieren und er wünscht daher eine Lesebestätigung, um sie ggfs. telefonisch zu befragen, falls das Feedback auf sich warten lässt. Er definiert als „Teilungsabsicht“, dass Kommentare zu diesem Dokument erwartet werden. Nach einer Bestätigung der Zusammenfassung seiner Teilungsabsichten kann Fred direkt weiteren Arbeiten nachgehen und wird von der Software darüber informiert, wenn der Upload abgeschlossen ist, das Dokument zur Verfügung gestellt wurde und seine Freunde es gelesen haben.

Freds Kollege Lukas liest und kommentiert als erster das Dokument. Fred erhält dazu einen Hinweis direkt auf seinem Computer und schaut sich den Kommentar in seinem Browser an. Dort sieht er auch, dass Pete das Dokument bereits erhalten und gelesen hat, Werner jedoch noch nicht. Er kommentiert Lukas' Anmerkungen direkt im Browser und ruft dann Werner an, um mit ihm über Lukas' Hinweise zu sprechen.

Am nächsten Tag erinnert die Software Fred daran, dass er gestern ein Dokument geteilt hat, dessen Ziele (durch Kontakte gelesen, Kommentare erwartet) erreicht

worden sind und ob es nun noch weiterhin für seine Kollegen verfügbar sein sollte oder aus dem Web gelöscht werden kann. Fred entscheidet sich dafür das Dokument auch weiterhin mit seinen Freunden zu teilen, da diese es für ihre Arbeit benötigen werden.

2.4.2.2 Maximilian Prange bittet um Telefonnummern

Um sein versehentlich gelöschtes Handy-Adressbuch zu rekonstruieren, möchte Maximilian Prange all seine Freunde bei Facebook anschreiben, um sie um die Zusendung ihrer aktuellen Telefonnummer zu bitten.

Gestern löschte Maximilian versehentlich alle Nummern aus dem Telefonbuch seines Mobiltelefons. Da er kein Backup davon besitzt, bleibt ihm aus seiner Sicht wohl keine Alternative, als seine Freunde um die Zusendung ihrer aktuellen Nummern zu bitten. Leider hat er mit den Telefonnummern auch die E-Mail-Adressen gelöscht, so dass ihm nur der Kommunikationsweg über das Web bleibt. Bei Facebook hat er viele Kontakte, die auch vorher in seinem Adressbuch standen. Daher entscheidet er sich dort nachzufragen.

Maximilian ist zuhause, klappt sein Notebook auf dem Küchentisch auf und startet den Web-Browser. Er navigiert zum Formular zum Erstellen einer neuen, privaten Facebook-Nachricht auf Basis der zu entwickelnden Architektur. Dort wählt er aus all seinen Kontakten mehrere Dutzend Empfänger aus und schreibt einen kurzen Text, in dem er die Situation erklärt und um die Zusendung der Nummern bittet. Er signiert die Nachricht, so dass seine Freunde sich auch sicher sein können, dass *er* — und niemand, der sich als Max ausgibt — sie um ihre vertraulichen Kontaktdaten bittet und stellt zudem ein, dass die gesamte Kommunikation vertraulich stattfinden soll. Dann sendet er die Nachricht ab und erhält eine Erfolgsbestätigung.

Wenige Minuten später erhält er die ersten Benachrichtigungen über Antworten auf seine Nachricht. In seinem Browser liest er sie und tippt die neuen Nummern direkt in sein Mobiltelefon. Danach archiviert er die Nachrichten als sicheres Backup für zukünftige Versehen.

2.4.2.3 Peer Maria Senfmann bewirbt sich um eine Wohnung

Um innerhalb der nächsten 14 Tage eine Wohnung zu finden, möchte Peer Maria Senfmann einem potentiellen Vermieter online seinen Lebenslauf, eine eingescannte Schufa-Auskunft und seine letzten drei Gehaltsabrechnungen im PDF-Format zur Verfügung stellen.

Um seine Promotion an der Universität des Saarlandes zu beginnen, zieht Peer derzeit von Kiel nach Saarbrücken. Die Wochenenden verbringt er dazu vollständig im Saarland auf der Suche nach passenden Wohnung. Nun hat er zwei Wohnungen, die ihm gefallen, in Aussicht gestellt bekommen und möchte seinen potentiellen Vermietern zügig erwünschte, persönliche Informationen über sich mitteilen, damit diese ebenso schnell eine Entscheidung treffen können. Peer hat etwas Zeitdruck, da er in 14 Tagen die Arbeit in Saarbrücken aufnehmen soll, die auch einen Lehrauftrag beinhaltet und somit Präsenz vor Ort erzwingt.

Peer befindet sich nach der letzten Wohnungssuche nun in einem Saarbrücker Café und hat sein Notebook auf den Knien platziert. Die notwendigen Unterlagen hatte er vorher schon vorbreitet und trägt sie digital bei sich. Er verbindet sich mit dem unverschlüsselten WLAN des Cafés und anschließend, um die Verbindung gegen *Sniffing*-Attacken abzusichern, mit dem VPN-Server der Universität. Dann startet der die Desktop-Software zur Nutzung der zu entwickelnden Architektur. Er öffnet die Ansicht zur Erstellung eines neuen Inhaltsobjektes und verfasst ein kurzes Anschreiben an den Vermieter. Dann fügt er diesem seine drei Dokumente hinzu und tippt die E-Mail-Adresse des Vermieters von seinem Schreibblock ab, auf der er sich während der Besichtigung Notizen gemacht hat. Peer markiert sämtliche Inhalte als vertraulich und erlaubt dem Empfänger die Nutzung ausschließlich zur Entscheidungsfindung in der Wohnungsfrage. Weitergabe und weitere Nutzungen sind untersagt. Ebenso vermerkt er als Ablaufdatum der Verfügbarkeit des Inhaltes „Montag in zwei Wochen“, da bis dahin die Angelegenheit geklärt sein sollte und sein Vermieter die Informationen dann aus seiner Sicht nicht mehr benötigen sollte. Er bestätigt die Auswahl und das Geschriebene und wartet die Teilungsbestätigung ab.

Der Vermieter erhält eine Benachrichtigung über den geteilten Inhalte zusammen mit Peers Anschreiben per E-Mail und kann die Anhänge im Browser ansehen und von dort aus drucken, jedoch nicht herunterladen oder weiterleiten.

Nach fünf Tagen erhält Peer die Benachrichtigung, dass der Vermieter die Inhalte zum ersten Mal angeschaut hat. Danach gibt’s jedoch kein weiteres Feeback von dessen Seite. Vermutlich hat er sich gegen Peer entschieden. Nach weiteren fünf Tagen nimmt Peer die Freigabe der Inhalte zurück, löscht sie allerdings noch nicht komplett aus dem Web, damit er sie ggfs. die weitere Wohnungsbewerbungen noch einmal nutzen kann. Sein Vermieter besitzt nun keinen Zugriff mehr auf die Informationen.

2.4.2.4 Michel „muck“ Langhanns möchte der Schnellste bleiben

Um auch weiterhin als schnellster Informationsverteiler in der Branche wahrgenommen zu werden, möchte Michel „muck“ Langhanns eine eben entdeckte dpa-Meldung direkt in einem neuen Tweet verlinken.

Es ist Anfang April 2010, muck sitzt vor seinem 27" iMac in der Agentur und surft durchs Web. Die Deutsche Presseagentur (dpa) meldet seit wenigen Minuten, dass die Bundesregierung beschlossen hat im Rahmen einer „Gestaltungsoffensive Deutschland 2011“ die Einkommensteuerpflicht freiberuflich arbeitender Grafikdesigner für das Jahr 2011 auszusetzen. muck ist begeistert und möchte dies umgehend seinen über 2000 Followern bei Twitter und auch bei identi.ca mitteilen, um für viele die Informationsquelle Nr. 1 im Bereich professionelles Grafikdesign zu bleiben.

Er kopiert dafür die Web-Adresse der dpa-Meldung aus seinem Browserfenster und navigiert zu einem Web-Frontend, das auf der neu zu entwickelnden Architektur arbeitet. Dort wählt er aus, eine neue Kurznachricht verfassen zu wollen, und anschließend die Services auf denen sie veröffentlicht werden soll: Twitter, identi.ca und auch in seinem privaten Weblog unter `http://muck.la`. Dann fügt er den Link ein und erhält zur Bestätigung der Echtheit der Informationen eine Auswertung des HTTPS-Zertifikats der dpa-Domain angezeigt. So kann er sich sicher sein, dass die Nachricht tatsächlich echt ist und er keinem *Phishing* aufgesetzt ist. Nachdem er dem Link einen kurzen Text hinzugefügt hat, veröffentlicht er die Kurznachricht und erhält für jedes ausgewählte Ziel-Netzwerk eine Erfolgsbetätigung. Nach einer kurzen Überlegung entscheidet sich muck diese Nachricht für sein Weblog etwas größer aufzuarbeiten und gegen die Veröffentlichung der Kurznachricht zu diesem Zeitpunkt an dieser Stelle. Er entfernt daher sein Weblog als Ziel der Nachricht, aktualisiert diese und so verschwindet die Kurznachricht wieder aus seinem Weblog.

Als ihn zwei Tage später ein Freund im Café trifft, fragt dieser ihn, ob der Feed seines Weblogs kaputt sei, da er dort vor kurzem einen Eintrag darin gefunden hätte, der kurze Zeit später schon nicht mehr da war. muck klärt seinen Freund auf, dass er den Beitrag wieder herausgenommen hätte, und ihm sogar die genaue Uhrzeit nennen, zu der er kurz in seinem Weblog verfügbar war. Denn diese ist weiterhin in der Nachricht dokumentiert.

2.4.3 Ermittelte Anforderungen

Zur Ermittlung der Anforderungen wurden nun bereits zahlreiche Aktivitäten durchgeführt: Eine umfangreiche Aufarbeitung der Problemdomäne ist in den Abschnitten

2.2.1 und 2.2.3 dokumentiert. Zudem wurden die in Abschnitt 2.2.2 gewonnen Erkenntnisse aus Benutzerbefragungen in den Abschnitten 2.3.1, 2.3.2, 2.3.3 und 2.3.4 in Modelle überführt. Auf dieser Basis sind zahlreiche Kontextszenarien entstanden und im Abschnitt 2.4.2 auszugsweise dokumentiert, die nun als weiterer *Input* unmittelbare Quellen der Anforderungen sind. Es gilt die in den Szenarien beschriebenen Aktionen, Objekte und Kontexte herauszustellen und diese auf funktionale, datenzogene und kontextbezogene Anforderungen abzubilden.

Im Rahmen der hier durchgeführten Anforderungsermittlung lassen sich die Anforderungen in fünf Themen-Cluster gruppieren:

- das Cluster „vertraulich agieren“ soll alle Anforderungen beschreiben, die in Zusammenhang mit der Gewährleistung von Vertraulichkeit beim Teilen eigener Inhalte stehen
- das Cluster „teilen und kontrollieren“ greift zusätzlich, aber bewusst getrennt davon, Anforderungen auf, die in Zusammenhang mit der Gewährleistung von Verfügbarkeit und Integrität eigener Inhalte im Web stehen; denn es stellt sich heraus, dass diese drei Konzepte durchaus getrennt voneinander betrachtet werden sollten
- das Cluster „vertrauen und überprüfen“ vereint alle funktionalen und datenbezogenen Anforderungen, die in Zusammenhang mit der Vertrauenswürdigkeit der Architektur stehen
- das Cluster „Bekanntes wahren und integrieren“ zielt auf die Integration bestehender Web-Anwendungen in die Architektur ab und die Erhaltung eingespielter Arbeitsabläufe
- das Cluster „Komplexität respektieren und reduzieren“ gruppiert schließlich die Anforderungen der potentiellen Nutzer an den Grad der akzeptablen Komplexität der Architektur

2.4.3.1 Bisherige Anforderungen

In vorangegangenen Abschnitten wurden teilweise bereits erste Anforderungen an die zu entwickelnde Architektur formuliert — teils noch sehr vage, teils schon konkreter —, die an dieser Stelle zur Schaffung eines Überblicks zusammengestellt sind. Im Abschnitt 2.2.1.1 (Vokabular, Teilen) war dies:

Eine Architektur zum Teilen von Inhalten im Web, die die Schutzziele der IT-Sicherheit bezüglich der teilbaren Inhalte berücksichtigt, sollte das unbegrenzte, digitale Teilen

ermöglichen, jedoch versuchen den Urhebern der Inhalte eine größtmögliche Kontrolle über ihre Inhalte zu bewahren. Die unautorisierte Weitergabe geteilter Inhalte sollte eingeschränkt werden.

und aus der darauf folgenden Betrachtung des Vertrauensbegriffs in Abschnitt 2.2.1.2:

Eine Architektur zum Teilen von Inhalten im Web, die als vertrauenswürdig empfunden werden soll, sollte ihren Aufbau und ihre Funktionsweise offen und nachvollziehbar kommunizieren und Möglichkeiten zur Überprüfung anbieten.

Aus der Überlegung heraus was „Nachvollziehbarkeit“ bedeute, entwickelte sich in Abschnitt 2.2.1.3 die Forderung:

Eine Architektur zum Teilen von Inhalten im Web, deren Alleinstellungsmerkmal „kontrolliertes Teilen“ für ihre Benutzer nachvollziehbar sein soll, sollte alle Schritte des komplexen Prozesses des Teilens deutlich an die Benutzer kommunizieren können. Denn Nachvollziehbarkeit bedingt das Vorhandensein aller notwendigen Informationen zu einem Vorgehen.

und alle dem steht das erste High-Level-Goal aus Abschnitt 2.1 als Prämisse voran:

Auf der zu entwerfenden Architektur basierend sollen Web-Anwendungen entwickelt werden oder bestehende Web-Anwendungen aufsetzen können [...]

Dies vorausgeschickt werden in den folgenden Abschnitten die weiteren konkreteren Anforderungen beschrieben, die sich v.a. aus der umfangreichen Auswertung der Benutzerbefragungen, der Betrachtung des Ökosystems und schließlich der Kontextszenarien ergaben. Anforderungen werden innerhalb der Cluster-Beschreibungen zur Verdeutlichung **fett** gedruckt.

2.4.3.2 Vertraulich agieren

Menschen besitzen das Bedürfnis Inhalte im Web im Vertrauen zu teilen. Niemand außer autorisierten Personen sollte dann Wissen davon oder Zugriff darauf erlangen. **Eine Architektur sollte ein vollständig vertrauliches Teilen ermöglichen, bei dem das Schutzziel der Vertraulichkeit vollständig zufrieden stellend erfüllt wird**. Eine Vorbedingung dazu ist die zweifelsfreie Identifizierung der kommunizierenden

Personen. Nur in einem bekannten und gesicherten Umfeld kann vertrauliche Kommunukation überhaupt stattfinden. **Eine Architektur sollte daher auch eine zweifelsfreie Identifizierung von Online-Kontakten ermöglichen**.

Menschen treten im Web-Kontext oft unter verschiedenen Identitäten parallel auf, den so genannten Personae[48]. Es kommunizieren in diesem Fall Personae untereinander, nicht die dahinter stehenden Menschen. Das Teilen eines Inhaltes findet im Web in einem *Kontext* aus Personae und deren Kontakten statt. **Eine Architektur sollte das Persona-Konzept berücksichtigen**. Eine Persona kann auch nur durch ein Pseudonym beschrieben sein.

2.4.3.3 Teilen und kontrollieren

Die Anforderungen dieses Clusters ergänzen die des ersten vor allem um die Aspekte bezüglich der Integrität und Verfügbarkeit geteilter Inhalte. Denn eine Grunderkenntnis aus den Befragungen, Modellierungen und Szenarien ist: Nicht immer wird die Trias aus Vertraulichkeit, Verfügbarkeit und Integrität für das Teilen im Web vollständig beansprucht. Es existiert vielmehr das Konzept einer *hinreichenden Absicherung*, die je nach Nutzungskontext und geteiltem Inhalt verschieden sein kann. Inhalte können öffentlich, also nicht vertraulich, geteilt werden, sollen aber trotzdem integer und verfügbar bleiben. Dass die Integrität im Web geteilter Inhalte durch andere Personen als den Urheber bewertet werden kann, ist in beinahe jedem Nutzungskontext erwünscht, eine Verfügbarkeit von Inhalten mindestens für deren Urheber ohne Zweifel auch. **Eine Architektur sollte kontextabhängige, Inhalts-individuelle Ansprüche an Vertraulichkeit, Verfügbarkeit und Integrität eigener Inhalte berücksichtigen und dem Benutzer die Modellierung seiner persönlichen, hinreichenden Absicherung erlauben.** Die Entkopplung der Erfüllung der Schutzziele funktioniert jedoch nicht beliebig: Integrität setzt Verfügbarkeit voraus und Vertraulichkeit setzt Integrität voraus. Diese Anforderung beinhaltet in der Form eine Schlüsselerkenntnis und wird im weiteren Designprozess eine wesentliche Rolle bei der konkreten Umsetzung der Architektur spielen.

Neben dieser grundlegenden funktionalen Anforderung, existieren zahlreiche weitere, teils nahe liegendere Anforderungen: **Die Architektur sollte weiterhin das Teilen mit mehreren Personen gleichzeitig erlauben, Kommentare auf geteilte Inhalte erlauben und geteilte Inhalte archivieren können.** Dieses Cluster umfasst allerdings auch Anforderungen, die das Ausüben von Kontrolle über geteilte Inhalte adressieren. Eine in diesem Zusammenhang grundlegende Anforderung ist, dass

[48] die zur besseren Unterscheidbarkeit zur Persona-Methode aus der *Modeling*-Phase stets mit dem lateinischen Plural geschrieben werden

die Architektur das Löschen eines geteilten Inhaltes respektive das „Rückgängig machen“ eines Teilungsvorgangs ermöglichen sollte. **Geteilte Inhalte** sollten daneben auch datenbezogene Anforderungen erfüllen, die von der Architektur beim Umgang mit Inhalten berücksichtigt und vom Urheber kontrolliert werden sollten: Sie **sollten**

- **Meta-Informationen über Nutzungslizenzen besitzen** (Darf ein Inhalt weitergegeben werden? Wenn ja, unter welchen Bedingungen?)
- **Meta-Informationen über ihre Lebensdauer besitzen, nach der sie nicht mehr im Web für ihre Teilhaber verfügbar sein sollten**
- **überprüfbar sein hinsichtlich ihrer Integrität**
- **aus dem Kontext der Architektur exportierbar sein, um auch in neuen Kontexten genutzt werden zu können, die die Architektur nicht berücksichtigt** (Gewährleistung von Verfügbarkeit der Inhalte über Architekturgrenzen hinweg)

Inhalte sollten darüber hinaus dem Inhaltsmodell aus Abschnitt 2.3.3 entsprechend, um primäre Payload- und sekundäre Meta-Informationen zu ihnen abbilden zu können.

2.4.3.4 Vertrauen und überprüfen

Zur Ausbildung wahrnehmbarer Vertrauenswürdigkeit der Architektur sind die drei bereits in vorangegangenen Diskussionen als grundlegend erachteten Anforderungen zu erfüllen (siehe auch Abschnitt 2.3.4): **Die Architektur sollte**

- **eine ausreichende Transparenz und gute Selbstbeschreibungsfähigkeit aufweisen**
- **eine Bewertbarkeit durch externe Autoritäten zulassen und ihre Bewertungen Interessenten kommunizieren können**
- **eine ausreichende Dokumentationsfähigkeit der in der Nutzung der Architektur gewonnenen Erfahrung von Benutzern besitzen, damit diese als Basis für die künftige Vertrauensbeziehung genutzt werden kann**

Dazu ergänzend kann in diesem Zusammenhang angeführt werden: **Die Architektur sollte**

- **die Konsequenzen des Handelns ihrer Benutzer stets vorweg nehmen und**

den Benutzer dadurch auf Basis aller dafür notwendigen Informationen Entscheidungen bewusst treffen lassen

- **sämtliche Vorgänge bei der Bearbeitung und Verwaltung nutzergenerierter Inhalte dokumentieren und auch im Nachinein noch nachvollziehbar und überprüfbar kommunizieren können**
- **nicht primär durch eine Person oder ein Unternehmen entwickelt und vermarketet, sondern als Communityprojekt etabliert werden**

2.4.3.5 Bekanntes wahren und integrieren

Dieser Themenbereich, der im Vorfeld der Benutzerbefragungen so nicht deutlich herausgestochen ist, nun aber als separater Punkt unbedingte Beachtung finden soll, adressiert die Benutzeranforderungen hinsichtlich der Integration aktueller Problemlösungen und die Berücksichtigung aktueller Workflows. Viele der befragten Personen und auch die modellierten Personas besitzen größtenteils bereits Accounts bei verschiedenen Social Networks und anderen Anwendungen, innerhalb derer man eigene Inhalte teilen kann. Dort existieren aktive Freundschaften, die nicht immer in neue Kontexte übernommen werden können. Eine neue Insellösung zu erschaffen, die isoliert vom Rest des Webs funktioniert, widerstrebt der Erreichung der erkannten Ziele (vgl. dazu auch Abschnitt 2.2.3.3). **Eine Architektur sollte daher auch die Aktivität ihrer Benutzer in bestehenden Social Networks berücksichtigen und Möglichkeiten bieten, für die dort geteilten Inhalte zumindest in Ansätzen die Erreichung der Schutzziele zu gewährleisten**.

Darüber hinaus findet das Teilen nicht selten *nicht* unter der Verwendung eines Web-Browsers statt (vgl. u.a. Szenario von Fred Busch), sondern per Client-Software, die entweder auf einem Dektop-Rechner oder auch mobil genutzt werden kann und sich auf Betriebssystemebene aus verschiedenen Gründen besser in Arbeitsabläufe eingliedern lässt als Web-Oberflächen. Neuartige Anwendungen, die einen eingespielten Arbeitsablauf ersetzen, haben es schwer von neuen Benutzern akzeptiert zu werden. **Eine Architektur sollte das Teilen via Client-Software ermöglichen und sich dadurch möglichst nahtlos in verschiedene etablierte Arbeitsabläufe integrieren lassen**. Für viele bestehende Social Networks existieren bereits Client-Anwendungen für unterschiedliche Betriebssysteme (iOS, Mac OS X, Windows, Linux, Android, WebOS ...). Es wäre wünschenswert, wenn Entwickler dieser Client-Software mit geringem Aufwand die Nutzung der alternativen Architektur ermöglichen könnten.

Eine völlig andere Perspektive, die jedoch unter der gleichen Überschrift eingenommen werden kann, ist **die Anforderung an die Integration bereits vorhandener**

(Teil-)Problemlösungen. Denn um im Wettbewerb mit bestehenden, wirtschaftlich betriebenen Social Networks freie Entwicklungsressourcen aus der meist ehrenamtlich arbeitenden „Community“ zu bündeln und nicht Initiativen mit ähnlichen Zielen parallel zu entwickeln, sollte eine Integration bestehender Systeme, Konzepte und Initiativen in die neue Architektur evaluiert werden (vgl. zu den Hintergründen dieser Anforderung Abschnitt 2.2.3.3).

2.4.3.6 Komplexität respektieren und reduzieren

Die letzte Gruppen aus Anforderungen geht schließlich auf die technische Ausgestaltung und Außendarstellung der Architektur ein. Während der Benutzerbefragungen wurde es deutlich und der „Unerfahrene“ Archetyp verkörpert es als herausstechendes Benutzermerkmal: Nicht alle potentiellen Benutzer besitzen ausreichendes Fachwissen, um komplizierte Software zu installieren und zu benutzen. Die neue Architektur soll jedoch diese breite Zielgruppe erreichen. **Eine Architektur sollte aus diesem Grund so einfach nutzbar wie möglich sein. Dies schließt auch die Installation und den Betrieb dazu benötigter Software mit ein**. Dass man durch aufwändige Systeme zwar technische Sicherheit gewährleisten kann, zeigen die in Abschnitt 2.2.3 beschriebenen Projekte. Diese zeigen jedoch auch, wie durch eine zu hoch angesetzte Latte bei der technischen Absicherung der Schutzziele Nutzungsbarrieren entstehen und nur noch eine stark verkleinerte Zielgruppe angesprochen werden kann. Diese Tatsache zu berücksichtigen und sich auch in der Außendarstellung nicht als „geeky solution“ zu bezeichnen, dies sollte eine weitere Anforderung an die Architektur sein: **Eine Architektur sollte für seine Nutzung nur ein geringes technisches Vorwissen voraussetzen und sich nicht als zu komplex im Markt positionieren, selbst wenn dies im Detail zustimmt**. Für diesen Fall sollten geeignete Kommunikationsmaßnahmen gefunden werden diese tatsächlich vorhandene Komplexität in der Kommunikation nach außen zu reduzieren — ohne dabei auf Transparenz zu verzichten.

2.5 Design Framework

In der *Design Framework*-Phase des Goal-directed Designs soll nun ein erster Schritt in Richtung eines konkreten Architekturentwurfes gemacht werden. Zum ersten mal wird der Frage *«Wie sollen die Anforderungen erfüllt werden?»* nachgegangen, wenn auch noch nicht auf einer finalen Detailebene.

2.5.1 Eine Architektur zum Teilen im Web

Maßgeblich beeinflusst wurde der Architekturentwurf von den folgenden Feststellungen. Die Ziele der Benutzer und die daraus ermittelten Anforderungen offenbaren: Eine vertrauenswürdige Architektur zum Teilen eigener Inhalte im Web kann *nicht* allein als eine technische Grundlage neuer Web-Anwendungen verstanden werden (und dadurch womöglich zur Bildung einer Insellösung beitragen), sondern es fehlt im Web ein ganzheitliches Konzept der Gewährleistung von Vertraulichkeit, Verfügbarkeit und Integrität nutzergenerierter Inhalte. Stellt man sich als Problemlösung eine neuartige Architektur als *Distributed Social Networking*-Anwendung (wie z.B. *diaspora**) vor, die das Teilen von Inhalten unter Absicherung der Schutzziele gewährleistet, aber dies eben nur für mit ihr erstellte Inhalte im Kontext dieser einen Anwendung tut, so blendet man den großen Rest des Webs mit seinen zahlreichen Vorkommen an nutzergenerierten Inhalten aus — und dies ist nicht im Sinne der Zielerreichung potentieller Benutzer.

Eine Architektur, die die formulierten Schutzziele gewährleisten kann und die sämtliche Benutzeranforderungen berücksichtigt, muss *web-weit* ausgerichtet sein, sich in das Gefüge aus bestehenden Web-Services einbetten, existierende Anwendungen über ihre Schnittstellen integrieren und dabei als universelles „Werkzeug-Kit“ zur Absicherung der Schutzziele begriffen werden, welches in verschiedensten Anwendungskontexten genutzt werden kann. Nicht die ständige, vollständige Erreichung aller Schutzziele steht im Vordergrund, sondern die sich aus einem Teilungskontext ergebende notwendige Erreichung einzelner Ziele. Die hierzu entwickelte Architektur soll jedoch weiterhin einfach kommunizierbar und nutzbar bleiben und auch den Anforderungen an Transparenz und Nachvollziehbarkeit gerecht werden.

Diese neue Denkrichtung äußert sich in der im folgenden Absatz beschriebenen Architekturskizze.

2.5.2 Vier Elemente bilden eine neue Architektur

Eine Architektur *für das Web* soll entwickelt werden. Diese besteht als solche aus mehr Teilen als nur einem Stück entworfener Software. Die Integration ins Web und die Aufforderung zur Nutzung der entwickelten Softwarelösung sind auch Bestandteile der Architektur. Als Problemlösung wird daher im Rahmen dieser Ausarbeitung die folgende Architektur aus vier Systemelementen zur Evaluation vorgeschlagen.

Manifest Da das Thema „Gewährleistung von Schutzzielen nutzergenerierter Inhalte“ offensichtlich noch nicht in der „Mitte des Webs“ angekommen und nicht als zentraler Arbeitsinhalt von Social Networks erkennbar ist, soll den Anfang der Architekturbeschreibung ein Manifest darstellen. Dieses öffentlich zugängliche Wenige-Punkte-Papier empfiehlt einerseits den Benutzer von Web-Anwendungen den verantwortungsbewussten Umgang mit eigenen Inhalten und beinhaltet auf der anderen Seite eine Selbstverpflichtung für die Betreiber von Web-Anwendungen sich mit dem Thema auseinanderzusetzen und gewisse Grundregeln beim Umgang mit nutzergenerierten Inhalten einzuhalten. Ziel ist es, eine begriffliche Basis zu schaffen, auf der im Web über die Themen Vertraulichkeit, Verfügbarkeit und Integrität nutzergenerierter Inhalte gesprochen werden kann, und durch die Selbstverpflichtung eines Anwendungsbetreibers zur Einhaltung der aufgestellten Regeln ein erstes Vertrauensverhältnis beim Teilen im Web zwischen Benutzer und Betreiber aufzubauen.

Denn — wie bereits mehrfach festgestellt — kann eine Kommunikation im Web vor allem dann nicht vertraulich stattfinden, wenn nicht alle beteiligten Systeme durch die Kommunizierenden kontrolliert werden. Da allerdings Web-Angebote in die Architektur integriert werden sollen, die auf diese Weise nicht kontrollierbar sind, soll im Zuge der Gewährleistung *individueller, hinreichender Sicherheit* (Benutzeranforderung) jener Verhaltenskodex aufgestellt werden, zu deren Einhaltung sich Betreiber verpflichten können. „Wir verpflichten uns, die Inhalte unserer Kunden nicht auf Schlüsselworte zu durchsuchen.“ Eine beispielhafte Aussage wie diese ist zwar nicht kontrollierbar, aber ihre Existenz bildet eine vollkommen andere Entscheidungsgrundlage für Benutzer, ob sie ein Web-Angebot nutzen oder nicht. Sie beeinflusst deren Einschätzung, ob eine Anwendung ihrer Vorstellung von *hinreichender Sicherheit* entspricht.

Eine ausgeprochene Selbstverpflichtung der Regeln des Manifests ist damit eine Chance für existierende Social Networks ihren Benutzern einen vertrauenswürdigen Umgang mit ihren Inhalten zu kommunizieren. Gelingt es, dass ein solches Manifest im Web an Aufmerksamkeit und Relevanz gewinnt, so kann dessen Einhaltung für Benutzer ein zentrales Bewertungkriterium für Vertrauenswürdigkeit werden.

Die weitere Ausgestaltung des Manifests wird in Abschnitt 2.6.1 beschrieben.

Architekturmuster/-empfehlungen Aufbauend auf dem gemeinsamen Verständnis der Schutzziele und auf der Selbstverpflichtung externer Web-Anwendungen verantwortungsbewusst und den Regeln des Manifests folgend mit bei Ihnen veröffentlichten nutzergenerierten Inhalten umzugehen, lässt sich der Architekturentwurf zum Teilen von Inhalten im Web fortführen. Er beinhaltet als Kernelemente Architekturmuster und -empfehlungen, welche die wesentlichen Systemkomponenten beschreiben, die benötigt werden, um eine technische Erreichung der einzelnen Schutzziele in unterschiedlichen Web-Kontexten zu gewährleisten. Eine ausführliche Beschreibung der Architekturmuster beinhaltet Abschnitt 2.5.3.

Schnittstellenspezifikation Ergänzend zur Sammlung aus Architekturmustern und -empfehlungen beschreibt eine Schnittstellenspezifikation die Kommunikation dieser Komponenten untereinander und mit den übrigen Services und Anwendungen im Web. Im Mittelpunkt steht hier die Idee einer *Managing API*, die Web-Anwendungen implementieren sollten damit über diese, eigene Inhalte in unkontrollierbaren Teilungskontexten aus einer vertrauenswürdigen Umgebung heraus verwaltet werden können.

Referenzimplementierung Eine konkrete Umsetzung der Architekturkomponenten und der *Managing API* wird in einer Referenzimplementierung gezeigt. Diese soll nicht als einzig denkbare Implementierung der Architekturskizze verstanden werden, sondern als *ein Weg* sie zu interpretieren. Die Implementierung dient dabei vor allem dazu, interessierten Nutzern ein *Hands-On*-Erlebnis mit den Funktionen der Architektur zu ermöglichen und durch diese Erfahrung Motivation zur eigenen Nutzung und Vertrauen in die Funktionsweise der Architektur zu wecken. Eine Beschreibung der Referenzimplementierung findet in Abschnitt 3 statt.

2.5.3 Architekturmuster/-empfehlungen

Im Sinne der Demokratisierung von Anwendungslogik im Web und dem dezentralen Gedanken konsequent folgend, soll der in dieser Ausarbeitung entstehende Architekturentwurf nicht alleinige Gültigkeit für sich beanspruchen und eine genaue Implementierung vorgeben. Es werden lediglich Architektur*muster* skizziert und *-empfehlungen* ausgesprochen, die dann in ihrer Implementierung durch Entwickler interpretiert werden können, in ihrer Zusammenstellung jedoch bereits auf

konzeptioneller Ebene eine Erreichung der Ziele potentieller Nutzer absichern. Ein Architekturmuster einzuführen, das bedeutet grundlegende Merkmale von Systemkomponenten zu zeichnen und deren Beziehung untereinander zu definieren. Für konkrete Implementierungen werden dann Empfehlungen gegeben.

2.5.3.1 Skizze

Aufbauend auf der Idee, Vertraulichkeit, Verfügbarkeit und Integrität voneinander zu entkoppeln und Benutzern die Gewährleistung einer individuellen Zusammenstellung dieser Schutzziele zu ermöglichen, bilden die folgenden Aussagen über die separate Erreichung der Schutzziele die konzeptionelle Basis der Architektur:

- **Integrität** ist für Urheber und Teilhaber eines geteilten Inhaltes überprüfbar, wenn ein Vergleich eines im Web geteilten Inhaltes mit einem Refereninhalt möglich ist, dessen Echtheit gesichert ist.
- **Verfügbarkeit** bedeutet, dass eigene Inhalte in Web-Anwendungen dem Urheber und Teilhabern in einem vereinbarten Zeitrahmen zur Verfügung stehen. Hauptprobleme bei dessen Gewährleistunge sind vor allem Downtimes, unzureichende Service-Level-Agreements oder die Einstellung des Betriebs von Web-Anwendungen ohne Exportmöglichkeit der eigenen Inhalte. Bei einer Referenzspeicherung eigener Inhalten zur Integritätssicherung obliegt auch die Gewährleistung der Verfügbarkeit der Inhalte unmittelbar derem Urheber. Im Web geteilte Inhalte sind dann bei Downtimes einer Web-Anwendung nicht innerhalb dieser Anwendung, jedoch bei der Referenzquelle des Urhebers verfügbar.
- **Vertraulichkeit** kann nur in vollständig selbst-kontrollierten Systemen gewährleistet werden. Hierzu sind Ansätze und Anwendungen des *Distributed Social Networking* zu wählen, die z.B. aus dem Bereich *Peer-2-Peer-Social Networking* entspringen können.

Die technische Abbildung dieser Aussagen wird durch zwei Web-Services geleistet, welche die Grundpfeiler der neuen Architektur bilden, jedoch auch ergänzt werden können und zur Erreichung aller Benutzerziele (vollständige Vertraulichkeit) auch *müssen*:

- der **Signed Content Storage** ist ein persönlicher, selbst-kontrollierter Speicherort eigener Inhalte im Web, der als Web-Service umgesetzt wird. Hier werden sämtliche geteilten Inhalte eines Urhebers als Referenz digital signiert persistent

gespeichert und sind dort unter gewissen Bedingungen öffentlich einsehbar. Die Echtheit der dort abgelegten Inhalte ist gesichert.

- zusätzlich sichert ein **Identity Provider**-Service die Identität der eigenen Personae und deren Kontakte im Web. Als Basis für jedes im Web stattfindene Teilen von Inhalten stellt der *Identity Provider*-Service u.a. Funktionen zur Verfügung, zwei Kommunikationspartner zweifelsfrei zu identifizieren und unter diesen dadurch anschließend auch eine vertrauliche Kommunikation stattfinden lassen zu können.
- Ergänzt werden können die zwei Web-Services durch verschiedene *Distributed Social Network*-Systeme (z.B. auch durch *diaspora**), die verschiedene Schnittstellen implementieren müssen, um die Funktionen der beiden Web-Services nutzen zu können.

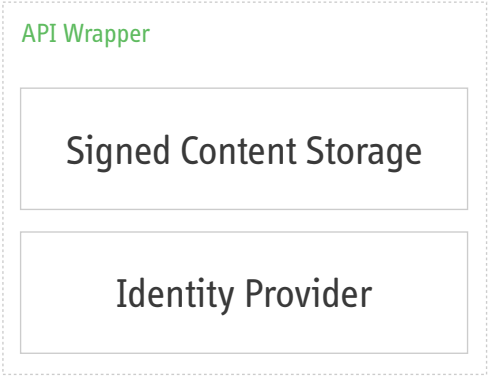

Abbildung 2.11: Grundpfeiler der neuen Architektur sind die Web-Services *Signed Content Storage* und *Identity Provider*.

2.5.3.2 Signed Content Storage

Der *Signed Content Storage* (SCS) adressiert in erster Linie die Forderung nach Verfügbarkeit und Integrität eigener Inhalte in unterschiedlichsten Web-Kontexten. Zudem erfüllt er die Benutzeranforderungen geteilte Inhalte dauerhaft archivieren zu können, verschiedene Meta-Informationen zu ihnen verfügbar zu machen, bereits in der Nutzung befindliche Social Networks auf einfache Weise in die Architektur zu integrieren und die Architektur über Client-Software in die eingespielten Arbeitsabläufe der Benutzer. Er ist in seiner Ausgestaltung auch eine Verkörperung von nachvollziehbarer Einfachheit und Transparenz und weist dedizierte Funktionen zur Nutzungserfahrungsdokumentation auf.

Wie der Name impliziert, greift der Service zur Gewährleistung von Integrität auf das Prinzip der *digitalen Signatur* von Inhalten zurück. Eine digitale Signatur soll in der digitalen Kommunikation die Echtheit und unverfälschte Übertragung eines Datums bestätigen und auch die tatsächliche Urheberschaft bezeugen. All dies erreicht der *Signed Content Storage*, indem er sich bei der Art und Weise der digitalen Signatur an den Gegebenheiten des Webs und den Teilungsmöglichkeiten innerhalb klassischer Social Networks orientiert: Im Gegensatz zur signierten E-Mail, die es aufgrund ihrer Spezifikation erlaubt, Zertifikate als Anhang mitzuverschicken, welche dann als Signatur gelten können, sehen die meisten Web-Anwendungen diesen Anwendungsfall so nicht direkt vor und bieten keine Möglichkeit derartige Informationen an eigene Inhalte anzuhängen. Der eigentliche Inhalt muss daher bereits die Signatur enthalten.

Beim *Signed Content Storage* erfolgt dies nach dem einfachen *Backlink*-Prinzip. Dadurch, dass ein Inhalt auf dem selbst-kontrollierten SCS-System öffentlich abrufbar gespeichert ist und zur Feststellung der Integrität ein Vergleich mit der Ursprungsversion des Inhaltes ausreicht, so genügt es, jedem geteilten Inhalt einen Link auf seinen Referenzinhalt hinzuzufügen. Bei Textinhalten kann dies unmittelbar im Text geschehen, bei Bildern kann dies in der Bildbeschreibung geschehen (sofern vorhanden), ansonsten auch in den Tags zu einem Bild oder sogar in den EXIF-Daten der Bilddatei. In visuellen Inhalten sind auch Wasserzeichen denkbar, in Audio-Inhalten kann man einen Link auch sprechen. Mit einem Inhalt assoziert gespeichert bekommt man einen Link in den meisten Fällen. Die genaue Ausgestaltung geschieht dann je nach Kontext[49].

Ein gewöhnlicher Tweet bei Twitter

@tmschndr: Ich denke über Integrität von Tweets nach.

wird zu einem mit einem Link signiertem Tweet.

@tmschndr: Ich denke über Integrität von Tweets nach. https://tim.sc/aS5fC

Der enthaltene Link öffnet eine HTML-Seite im Browser mit Detailinformationen zum Inhalt: dem originären Inhalt und seinen Metadaten. Hätte Twitter als Betreiber des Dienstes diesen Tweet verfälscht, würde es einem Betrachter der Detailseite nun auffallen. Die Domain *tim.sc*[50] gehört laut WHOIS-Eintrag dem Besitzer Tim

[49] bei Twitter wird ein Link vorzugsweise sehr kurz sein müssen, um nicht zu viele Zeichen zu beanspruchen

[50] die Top-Level-Domain *.sc* darf in diesem Fall auch als *Signed Content* interpretiert werden

Schneider, der sich für die Registrierung bereits beim Registrar ausgewiesen hat. Durch ein HTTPS-Zertifikat kann die Echtheit der dort abrufbaren Inhalte zusätzlich bestätigt werden. Dies sichert die Echtheit der Informationen zu einem Inhalt. Wenn eine Person ankündigt, dass sie fortan nur noch signierte Inhalte teilt, ist sie damit auch gegen Identätsmissbrauch geschützt, da „Doppelgänger“ keine derart signierten Inhalte auf der Domain erstellen können.

Neben der Signierung eigener Inhalte bietet der *Signed Content Storage* an, diese mit Metadaten anzureichern und erfüllt damit weitere datenbezogene Anforderungen an die Architektur. Definierbare Metadaten sind:

- sämtliche „Orte“ im Web mit deren **Permalinks**, bei denen dieser Inhalt in einem Teilungskontext abrufbar ist; dies ist besonders dann interessant, wenn ein Inhalt in mehreren Kontexten gleichzeitig geteilt wurde, z.B. bei Twitter und identi.ca. Der Permalink zurück auf den Inhalt in die autorisierten Kontexte ist eine zweite Möglichkeit zur Überprüfung der Integrität. Denn nur in den hier angegebenen Kontexten sollte ein Inhalt im Web auffindbar sein. Erscheint er woanders, ist diese Veröffentlichung nicht durch den Urheber autorisiert.
- eine maschinenlesbare **Nutzungslizenz**, die u.a. die Weitergabe oder kommerzielle Nutzung der Inhalte regeln kann
- eine maschinenlesbare, vom Urheber festgelegte **Lebensdauer**

Die Festlegung einer Lebensdauer, nach der ein Inhalt aus den Teilungskontexten wieder gelöscht werden sollte, gibt dem Urheber weitere Kontrolle über seine Inhalte, die in unkontrollierbaren Web-Anwendungen gespeichert werden. In der Anforderungsermittlung hat sich herausgestellt, dass nicht alle Inhalte ewige Relevanz besitzen, sondern auch nur zeitweise geteilt werden sollen. Eine Definition dieser Vorgabe zum Zeitpunkt des Teilens beugt dem Fall vor, Inhalte später doch nicht zu löschen. Natürlich gilt auch in diesem Punkt: Eine absolute Sicherheit, dass Inhalte nach dem Löschdatum tatsächlich aus fremden Web-Anwendungen gelöscht werden, kann allein die Angabe des Löschwunsches nicht gewährleisten, aber durch die Selbstverpflichtung des Manifests wird angenommen, dass sich Web-Anwendungen tatsächlich daran halten.

Allein der bisher beschriebene Funktionsumfang — Inhalt wird erstellt, Inhalt wird signiert, Inhalt ist öffentlich abrufbar — erfüllt bereits zahlreiche Benutzeranforderungen, bei denen es in erster Linie darum geht Inhalte einfach (aber unter Absicherung von Integrität und Verfügbarkeit) im Web verfügbar zu machen, z.B. um Dokumente zu teilen. Durch eine einfache Kommentarfunktion für Betrachter eines abrufbaren Inhaltes wird dieses Nutzungsszenario abgerundet. Durch die Nutzung der

Schnittstelle des Web-Services werden weitere Anforderungen erfüllt.

2.5.3.3 Identity Provider

Der *Signed Content Storage* kann auf einfache Art und Weise die Integrität und Verfügbarkeit von Inhalten gewährleisten, allerdings erfüllt er noch keine Anforderungen an die Echtheit der Kommunikationspartner und an eine komplett vertrauliche Kommunikation. Dieser nähert sich der *Identity Provider*-Web-Service an. Als eine auf das Wesentliche reduzierte Web-Anwendung ermöglicht er grundsätzlich das Modellieren von Personae und das Organisieren von Kontakten. Die Echtheit der Kontakte einer Persona kann dabei gesichert werden. Ziel ist es, autorisierte Informationen auch zur Identität von teilenden Personen/ae im Web verfügbar zu haben.

Diese Informationen können dann zum Beispiel vom *Signed Content Storage*, aber auch von *Distributed Social Networks* wie *diaspora** genutzt werden, um Inhalte in vertraulichen Kontexten zu teilen. Dieser Web-Service soll damit einen Gegenentwurf darstellen zur verstreuten Modellierung von Identitätsinformationen in isolierten Web-Anwendungen. Natürlich funktioniert eine Modellierung auch in diesen Netzwerken, doch ist dies aus der Idee heraus, vollkommene Kontrolle über eigene Inhalte, und damit auch über Informationen zur eigenen Person, zu gewährleisten, sinnvoll auch dieses grundlegende Beziehungsgepflecht einer Person auf einem vollständig selbst-kontrollierten System stattfinden zu lassen.

Die weitere Ausgestaltung des *Identity Provider*-Web-Services ist dem Abschnitt 2.6.3 zu entnehmen.

2.5.3.4 Gewährleistung von Vertraulichkeit

Bisher sichern die Komponenten der skizzierten Architektur die Gewährleistung von Integrität und Verfügbarkeit für öffentlich geteilte Inhalte im Web und legen mit gesicherten Informationen zur Identität der teilenden Personen die Grundlage für vertrauliches Teilen im Web. Nicht alle im *SCS* gespeicherten und signierten Inhalte sind per se öffentlich zugänglich und auch wenn sie dies sind, so bleibt durch den Einsatz öffentlicher, aber kryptischer Prüfsummen-URLs (siehe Abschnitt 2.6.2) die Anforderung an „hinreichende Sicherheit“ zunächst gewahrt, doch berücksichtigt die Architektur bisher keine Zugriffsbeschränkung auf geteilte Inhalte in der Form, dass nur autorisierte Personen auf sie zugreifen können. Hierzu ist eine Erweiterung der beiden Services notwendig.

Wie bereits festgestellt kann vertrauliches Teilen nur in *Distributed Social Networks* stattfinden, d.h. in dezentral organisierten Netzwerken. Anliegen dieser Initiative ist jedoch aus den in der Anforderungsermittlung genannten Gründen *nicht* allein das umgangssprachliche *Yet Another Distributed Social Network* als Insellösung zu entwerfen, sondern brauchbare (Teil-)Problemlösungen in eine neue, web-weite Architektur zu integrieren. Die Nutzung der beschriebenen Architektur als *Distributed Social Network* erfolgt erst im zweiten Schritt, ist jedoch möglich wie Abschnitt 2.6.2 darstellt. Daneben können auch andere *Distributed Social Networks* integriert werden: *diaspora** löst das Problem vertraulicher Kommunikation, zumindest auf konzeptioneller Ebene. Auch wenn dessen Umsetzung derzeit noch nicht vollständig durchgeführt ist, so werden sich verschiedene Social Networks auftun, die vertrauliche Kommunikation ermöglichen. Diese sichern dann vielleicht nicht primär die Gewährleistung von Integrität oder Verfügbarkeit, aber die von Vertraulichkeit. Die Kombination jener Netzwerke mit den Funktionen der bereits beschrieben Architektur führt zum Ziel die individuelle Gewährleistung einzelner Schutzziele in den Mittelpunkt zu stellen.

Über die in Abbildung 2.11 angedeutete und in den Abschnitten 2.5.4 und 2.6.2 beschriebenene Programmierschnittstelle findet diese Integration statt.

2.5.3.5 Schaffung von Vertrauenswürdigkeit

Nachdem die Grundlagen gelegt sind, um die Schutzziele der IT-Sicherheit technisch zu gewährleisten, gilt es diese Gewährleistung nachvollziehbar zu gestalten und u.a. dadurch die Vertrauenswürdigkeit der Architektur zu erschaffen.

Die drei in der Anforderungsermittlung genannten Punkte — ausreichende Transparenz und Selbstbeschreibungsfähigkeit, Bewertbarkeit durch Autoritäten und die ausreichende Dokumentationsfähigkeit eigener Erfahrungen — werden unmittelbare Bestandteile der Architektur.

Transparenz, Nachvollziehbarkeit und Selbstbeschreibungsfähigkeit werden erreicht durch:

- eine zentrale Projekt-Webseite, die sämtliche der folgenden Informationen beinhaltet
- eine offene Architektur, die sämtliche verwendete Software im Quellcode verfügbar macht oder darauf verweist und die Weiterentwicklung unter den Augen und im Diskurs mit der Web-Community forführt
- eine einfache Sprache zur Beschreibung des Aufbaus der Architektur unter Zuhilfenahme von Metaphern und einer Anleitung zur Exploration der gesamten

Architektur; hierdurch sollen Interessierte angeleitet werden ihr Verständnis der Architektur in der Tiefe zu verbessern

- das Bereitstellen von Screencasts, in denen die Funktionsweise der Architektur im Überblick erläutert wird sowie die Installation der zur Nutzung notwendigen Software, in denen aber auch die Durchführung einzelner Anwendungsfälle multimedial aufbereitet wird; durch abstrahierende, animierte Illustrationen lassen sich auch komplexere Systembeziehungen nachvollziehbar kommunizieren

Eine **Bewertbarkeit der Architektur** durch externe Autoritäten wird erreicht durch die Möglichkeit:

- über die Projekt-Webseite Meinungen, Kommentare und Erfahrungsberichte aus der Nutzung der Architektur zu veröffentlichen; diese können dann mit anderen Interessenten kritisch diskutiert werden und erscheinen ggfs. auch als *Testimonials* prominent auf der Projekt-Webseite, um Interessenten die Möglichkeit zu geben, sich anhand dieser Informationen einen Eindruck der Vertrauenswürdigkeit zu machen

- in der Nutzung der beschriebenen Dienste selbst die Zufriedenheit mit diesen aus Benutzersicht stets bewerten zu können; auch diese Meinungen können anschließend auf der Projekt-Webseite veröffentlicht werden oder in einen allgemeinen öffentlich einsehbaren *Score* einfließen, der angibt wie viel Prozent aller Nutzer derzeit mit der Architektur zufrieden sind

Die **Dokumentationsfähigkeit eigener Erfahrungen** wird erreicht durch eine kontinuierliche Evaluation der Architektur während der Nutzung: Die mit den Web-Services durchführbaren Aktionen sind bekannt und endlich. Es werden diejenigen zur Evaluation ausgewählt, die einem Benutzer besonderes Vertrauen abverlangen (z.B. das Veröffentlichen eines Inhaltes oder das entfernte Löschen eines Inhaltes aus einem angeschlossenen Social Network). Für diese wird während der Nutzung zum Zeitpunkt *vor* der Durchführung der Aktionen genau beschrieben welche Konsequenzen sie hat und wie diese im Anschluss überprüft werden können. Beispielsweise kann dies für die entfernte Löschung eines Inhaltes bedeuten:

Mit einem Klick auf den Button „Inhalt löschen“ löschen Sie den Inhalt X aus den Kontexten Y (Permalink: ...) und Z (Permalink: ...). Möchten Sie dies wirklich tun?
...
Sie haben den Inhalt X aus den Kontexten Y und Z gelöscht. Klicken sie die ehemaligen Permalinks an, um nachzuschauen, ob der der Inhalt tatsächlich gelöscht wurde. Bewerten

sie den Erfolg dieser Aktion: Das Löschen hat funktioniert, es hat nicht funktioniert.[51]

Die eigene Bewertung wird gespeichert und zukünftig vor dem Durchführen weiterer Aktionen angezeigt. Auf diese Weise wird nicht nur die eigene Erfahrung mit der Architektur dokumentiert, so dass man sich bei späteren Aufrufen der gleichen Aktion sicherer in seiner Erwartungshaltung sein kann („dieser Aktion hast du bereits 16 Mal vertraut"), man steigert durch dieses Vorgehen auch das Verständnis der Funktionsweise beim Benutzer, schult ihn damit beim Herleiten komplex erscheinender Vorgänge und befähigt ihn zur öffentlichen Bewertung der Architektur, z.B. in Form eines Testimonials, um damit anderen Interessierten bei ihrer Vertrauensbildung zu helfen.

2.5.4 Schnittstellenspezifikation

Abbildung 2.11 zeigt die beiden Web-Services *Signed Content Storage* und *Identity Provider* umschlossen von einem *API Wrapper*, d.h. sie sind ansprechbar für andere Web-Services über eine Programmierschnittstelle. Diese auf dieser Seite der Kommunikation zu definieren, aber auch auf der Seite bei Social Networks, die die Schnittstelle der beiden Web-Services nutzen möchten, dies ist Inhalt der Schnittstellenspezifikation als Teil der Gesamtarchitektur.

Die Idee ist es einerseits signierte Inhalte aus dem SCS für Web-Services über die API abrufbar zu machen und Informationen über Personae maschinenlesbar zugänglich zu machen. Auf der anderen Seite zielt die Schnittstellenspezifikation auch auf die Ausgestaltung der Programmierschnittstelle entfernter, angeschlossener Web-Services und -Anwendungen. Das Manifest verpflichtet diese zur Einhaltung von Benutzeraufforderungen (z.B. der Löschung eines Inhaltes) und diese auszulösen soll unmittelbar über ein Web-Frontend des *Signed Content Storage* möglich sein, d.h. es wird über die Schnittstellen der angeschlossenen Systeme verarbeitet. Dienste wie Facebook, die aktuell bereits eine umfangreiche Programmierschnittstelle anbieten und möglicherweise Interesse daran haben, die Grundaussage des Manifests zu unterstützen, sollten SCS-Systemen die Möglichkeit eröffnen, über ihre Programmierschnittstelle dort veröffentliche Inhalte auszulesen und auch zu verwalten (*read/write*-Zugriff auf Inhalte). Diese Schnittstelle wird im Folgenden als *Managing API* bezeichnet.

Wie die Referenzimplementierung es zeigt und die Anforderungen es vorschreiben, soll es auch möglich sein die Inhalte des *Signed Content Storage* über ein Web-Frontend

[51] die Unterstreichung soll einen Link kennzeichnen

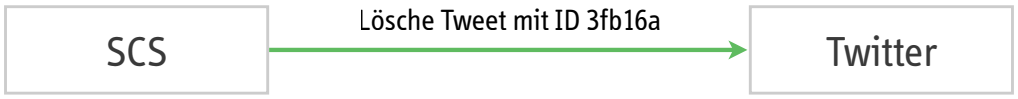

Abbildung 2.12: Entfernte Löschung eines Tweets über die *Managing API*.

oder über Client-Software zu erstellen, zu verändern und sie von dort aus auch in mehreren Social Neworks gleichzeitig zu veröffentlichen. Die hierzu notwendige Schnittstelle der vertrauenswürdigen Architektur wird als *Trusted API* bezeichnet.

Schnittstellenbeschreibungen sowohl für die *Trusted API*-Schnittstelle der Architektur (*Signed Content Storage*- und *Identity Provider*-Service), als auch die Anforderungen an Schnittstellen externer Web-Services (*Managing API*) sind im Anhang A.3 ersichtlich. Über die technische Gestaltung enthält auch der Abschnitt 2.6.2 weitere Informationen.

2.6 Design Refinement

Die Grundlagen der neuen Architektur sind in der Dokumentation der *Design Framework*-Phase beschrieben. Zwei Web-Services dienen der Gewährleistung von Verfügbarkeit und Integrität und ermöglichen die Nutzung weiterer Systeme zur Erreichung von Vertraulichkeit. Die Ausgestaltung dessen schafft zudem Vertrauenswürdigkeit. In der *Design Refinement*-Phase werden sämtliche Schlüsselkomponenten der Architektur noch einmal auf- und auf Detailebene ausgearbeitet. Dazu haben im Vorfeld im Rahmen einer formativen Evaluation Gespräche mit Domänenexperten stattgefunden, Testungen der skizzierten Architektur innerhalb der Gruppe potentieller Nutzer im Stile eines *Cognitive Walkthrough* und erneute Abgleichungen mit den modellierten Personas und deren Zielen. Die Ergebnisse werden in einer Reihe aus „Fokus"-Betrachtungen dargestellt. Diese sind ergänzend zu den Beschreibungen der *Design Framework*-Phase zu verstehen und ergeben zusammen mit diesen die vollständige Beschreibung der Architektur.

2.6.1 Fokus: Manifest

Die textuellen Inhalte des in Abschnitt 2.5.2 eingeführten Manifests sind nach Durchführung formativer Evaluation unter Zuhilfenahme der Personas und Gesprächen mit Domänenexperten zunächst die folgenden:

- eine Präambel, die an den bewussten Umgang mit persönlichen Inhalten im Web appelliert
- eine Selbstverpflichtung der Aussage: „Als Betreiber einer Web-Anwendung verpflichte ich mich, die Inhalte meiner Nutzer ihren Wünschen nach zu behandeln. An mich herangetragene Wünsche hinsichtlich der Löschung oder Veränderung von Inhalten werden erfüllt."
- eine Erweiterung dieser grundlegenden Erklärung um den Punkt der *Managing API*: „Als Betreiber einer Web-Anwendung stelle ich eine *Managing API* zur Verfügung, über die in meinem System gespeicherte Inhalte erstellt, gelöscht oder verändert werden können."

Die Zustimmung zu diesen beiden Aussagen ist für eine Unterzeichnung des Manifests unabdingbar. Darüber hinaus werden jedoch zur Einschätzung der durch den Betreiber verwendeten Architektur aus Benutzersicht verschiedene Erfüllungsgrade an gewährleisteter Vertraulichkeit definiert, die eine Web-Anwendung erreichen kann:

- *„basic“* kennzeichnet das Vorhandensein einer *Managing API* und einer mit SSL/TLS verschlüsselten Datenübertragung zwischen Client (Browser) und Server
- *„fair“* kennzeichnet das Vorhandensein einer *Managing API*, einer mit SSL/TLS verschlüsselten Datenübertragung sowie einer verschlüsselten Datenspeicherung (der Betreiber besitzt dadurch keinen inhaltlichen Zugriff auf Nutzerinhalte)
- *„full“* kennzeichnet das Vorhandensein einer *Managing API*, einer mit SSL/TLS verschlüsselten Datenübertragung, einer verschlüsselten Datenspeicherung sowie das Nicht-Vorhandensein von Log-Files (der Betreiber besitzt daher keine Information über Aktivitäten der Benutzer)

In diesem Punkt können unterzeichnende Betreiber eine Selbsteinschätzung treffen und die von ihnen getroffenen Vorkehrungen zum Schutze der Vertraulichkeit ihrer Nutzer deutlich machen (und durch diese Transparenz auch Vertrauenswürdigkeit schaffen).

Das Manifest umfasst (auch der Benutzerbarkeit wegen) tatsächlich nur diese wenigen Punkte. Allein die Existenz genügt im Prinzip, um damit ein Papier in den Händen halten zu können, auf dessen Aussagen man sich verpflichten kann. Die Kürze sorgt auch dafür, dass es sich leicht kommunizieren lässt — vor allem an Personen, denen es unbekannt ist. Hierbei kann auch eine grafische Umsetzung der Aussagen und Erfüllungsgrade beitragen, ähnlich wie die *Mozilla Privacy Icons* (siehe Abbildung 2.5) Datenschutzerklärungen visualisieren. Diesem Set, welches bereits Aufmerksamkeit auf sich gezogen hat, weitere Symbole mit den Aussagen des Manifests hinzuzufügen erscheint sinnvoll (siehe Abbildung 2.13).

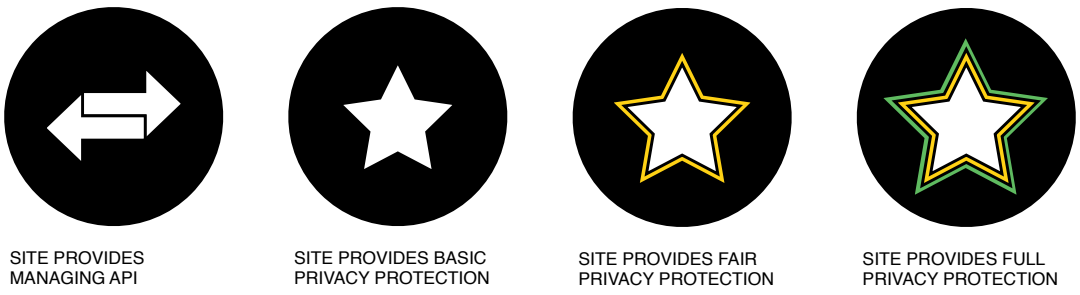

Abbildung 2.13: Erweiterung der Mozilla Privacy Icons um die Aussagen des vorgeschlagenen Manifests.

2.6.2 Fokus: Signed Content Storage

Auch das Konzept des *Signed Content Storage* wurde nach Gesprächen mit Domänenexperten und potentiellen Nutzern verfeinert. Zusätzliche Domänenrecherche festigte zudem die technischen Anforderungen, die durch Client-Software an den Service gestellt werden.

URL-Gestaltung Während das Backlink-Prinzip beibehalten wird, kann der Link allein dadurch weitere Benutzeranforderungen erfüllen, indem er direkt die SHA1-Prüfsumme des Inhaltes beinhaltet. Diese war zunächst nur auf der Detailseite zu einem Inhalt im *Signed Content Storage* als Metadatum vorgesehen (vgl. Abbildung 2.17), durch eine Nutzung unmittelbar *als* Link können geübte Anwender die Echtheit bereits überprüfen bevor sie die Seite aufgerufen haben, allein indem sie sich die SHA1-Summe des Inhaltes errechnen und sie mit dem Link vergleichen. Während dies tatsächlich eher nach einem Nutzungszenario für Domänenexperten klingt, kann dadurch auch ein Nutzen für den unerfahrenen Anwender entstehen: Denn die Platzierung der Prüfsumme bereits in der URL würde die Entwicklung von „Integritäts-Prüf-Software“ erleichtern, die auf diesem Konzept aufsetzt. So könnten beispielsweise Twitter-Clients entwickelt werden, die SCS-Links erkennen, die Prüfsummenberechnungen und Vergleiche automatisch durchführen und ihren Benutzern Abweichungen zwischen dargstelltem Tweet und Original melden. Ein solch automatisiertes Verfahren würde die Sicherung der Integrität auf ein neues Level heben.

OAuth als Schnittstellen-Autorisierung Der *Signed Content Storage* soll per *Trusted API* für Client-Software ansprechbar sein sowie für Social Networks, in denen Inhalte veröffentlicht werden sollen oder die Metadaten zu veröffentlichen Inhalten beim SCS abfragen sollen. Hierzu bedarf es zwei Arten von Autorisierungen: Client-Software muss die Erlaubnis erhalten auf die Inhalte im SCS zuzugreifen und neue zu erstellen und der Storage-Service selbst muss die Erlaubnis erhalten Inhalte im Namen des Benutzers bei den Social Networks einzustellen oder zu verändern. Zu diesem Zweck findet das offene Autorisierungsprotokoll OAuth[52] Verwendung. Bei der Verwendung von OAuth stellt ein Benutzer einer einer Anwendung A, die Zugriff auf Inhalte einer Anwendung B wünscht, *nicht* die eigentlichen *Credentials* (z.B. Benutzername und Passwort) von B zur Verfügung, mit der die Anwendung A sich gegenüber B authentifizieren könnte, sondern an B wird eine Nutzungsanfrage

[52]Beschreibung siehe http://oauth.net/, Spezifikation siehe http://tools.ietf.org/html/rfc5849

der Anwendung A gestellt, die ein Benutzer dort bei B bestätigen kann. Dann wird für die Anwendung A ein einmaliger, eigener *Token* erstellt, mit der diese sich bei B im Namen des Benutzers authentifizieren und Aktionen durchführen kann. Diese Vorgehensweise schützt Benutzername und Passwort eines Benutzers und gibt ihm damit Kontrolle über die autorisierte Durchführung von Aktionen in seinem Namen. Er hat jederzeit die Möglichkeit die Autorisierung von A bei Anwendung B zu widerrufen. Anwendung A kann dann keine Aktionen mehr in seinem Namen durchführen — ohne dass der Benutzer Anwendung A *Credentials* wegnehmen müsste oder sein Passwort ändern müsste, weil dies der Partei nun bekannt wäre.

OAuth ist in modernen Web-Anwendungen zum Quasi-Standard für API-Autorisierungen gereift und viele Web-Anwendungen empfehlen die ausschließliche Nutzung von OAuth in Zusammenhang mit Client-Software. Auch der *Signed Content Storage* soll aus diesem Grund OAuth an zwei Stellen einsetzen: Clients sollen auf diese Weise über die *Trusted API* Zugriff erlangen auf die Inhalte innerhalb des SCS erlangen können und der Service selbst soll per OAuth autorisierten Zugriff auf entfernte Inhalte über die Managing API angeschlossener Web-Anwendungen erlangen.

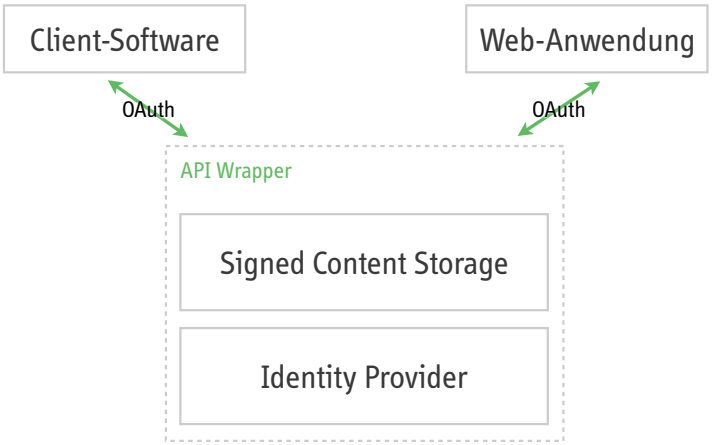

Abbildung 2.14: OAuth dient als Schnittstellen-Autorisierung der Architektur

Absicherung der Echtheit des Signed Content Storages Die Echtheit der beim *Signed Content Storage* abrufbaren Inhalte ist unbedingt sicherzustellen. In der Framework-Beschreibung wurde hierzu bereits die einfache, aber wirksame Idee entwickelt dies an die Echtheit der Domain zu knüpfen. Denn hierfür existieren bereits Zertifikat-Lösungen, so dass den Inhalten einer Domain vertraut werden kann und *Phishing* vorgebeugt wird. Dies funktioniert jedoch nur soweit gut, solange der Service auf einer eigenen Domain betrieben werden kann und wider-

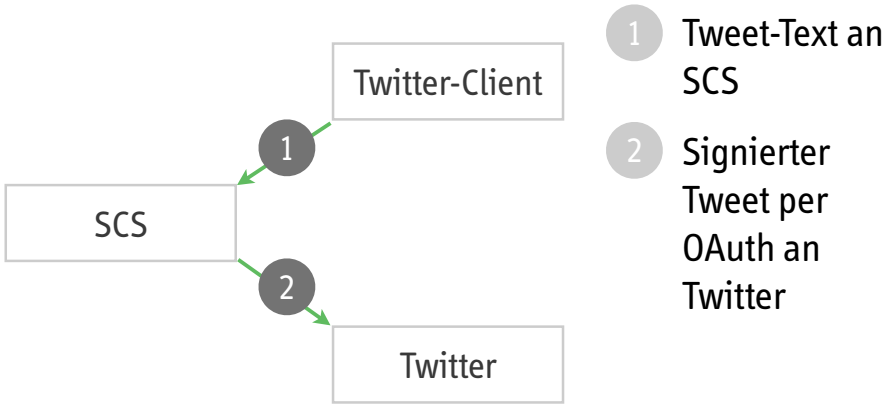

Abbildung 2.15: Implementiert ein Web-Service die OAuth-Autorisierung, so kann der *Signed Content Storage* direkt mit ihm kommunizieren.

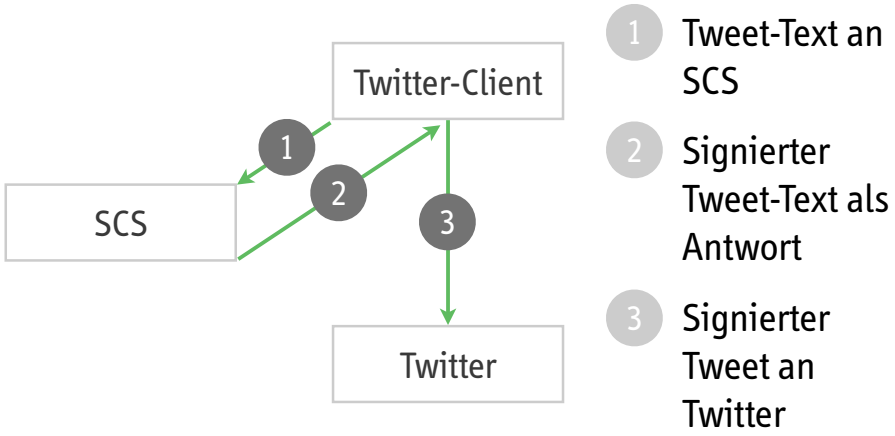

Abbildung 2.16: Implementiert ein Web-Service die OAuth-Autorisierung nicht, so kann der *Signed Content Storage* dennoch indirekt genutzt werden.

spricht der Anforderung an eine einfache Installation und Nutzung. Es ist davon auszugehen, dass größere SCS-Systeme existieren werden, die für die Nutzung mehrerer Benutzer bestimmt sein sollen, z.B. mit Backlinks in der Form http://scs.ccc.de/timschneider/ah3fGs1, bei denen der Domaininhaber der Domain *ccc.de* der Betreibers des Systems ist und die Benutzerzugehörigkeit eines Inhaltes aus der URL erst nach dem Domainnamen deutlich wird. In diesem Falle versagt die Absicherung der Echtheit anhand der Domain und eine neue Lösung wird benötigt.

All so genannter *Fallback* wird für diesen Fall eine Adaption des *Web of Trust*-Konzeptes vorgeschlagen[53]. Ein *Signed Content Storage* als Multiuser-System sollte eine öffentliche Profilseite der jeweiligen Persona besitzen, z.B. http://scs.ccc.de/timschneider. Auf dieser sollen andere SCS-Nutzer Bewertungen hinterlassen können, in denen sie mitteilen, ob sie die Echtheit des Profils verifizieren können und wenn ja aus welcher Annahme. Diese Bewertungen werden widerum per Backlink auf das andere SCS-System gegen Fälschung abgesichert. Sollte dieses SCS-System dann kein Multiuser-System sein, sondern sollte hier eine Überprüfung der Echtheit auf Basis der Domain möglich sein, so kann von einem Betrachter der Situation im Sinne des *Web of Trust* das Vertrauen in die Echtheit des bewerteten Profils übernommen werden. Dieses System funktioniert, solange genügend SCS-Systeme existieren, deren Echtheit an objektiv vertrauenswürdigen Kriterien festgemacht werden kann.

Metadaten Als Metadaten zu einem Inhalt sollen öffentlich abrufbar sein: Permalinks auf veröffentlichte Inhalte, Nutzungslizenz und Lebensdauer. Die Lebensdauer wird in Tagen angegeben und bei der Erstellung eines neuen Inhaltes kann auf Voreinstellungen für verschiedene Inhaltstypen zurückgegriffen werden. Die Auswahl der Nutzungslizenzen umfasst neben dem Copyright, welches jegliche Zweitnutzung untersagt, in erster Linie sämtliche Creative-Commons-Lizenzen[54], da diese bereits im Web erprobt, anerkannt und den Nutzern bekannt sind. Diesen berücksichtigen zudem bereits die Regelung der Weitergabe von Informationen und damit den Hauptzweck dieser Funktion in der Architektur. Ein Benutzer hat allerdings auch die Möglichkeit einen eigenen Lizenztext zu veröffentlichen, um individuelle Weitergaben zu bestimmen.

[53] Das *Web of Trust* ist in der Kryptologie die Idee, die Echtheit von digitalen Schlüsseln durch ein Netz von gegenseitigen Bestätigungen zu sichern. Es stellt eine dezentrale Alternative zu einer hierarchischen Public-Key-Infrastruktur dar, die autorisierte Aussteller von Schlüsseln benötigt. Für weitere Informationen siehe u.a. http://www.rubin.ch/pgp/weboftrust.de.html

[54] http://creativecommons.org

Abbildung 2.17: Beispiel einer Detailseite eines im *Signed Content Storage* gespeicherten Inhaltes.

Sicherung der Inhalte Um die Verfügbarkeit der Inhalte zu gewährleisten, sollten Implementierungen des *Signed Content Storage* zur Auflage haben, regelmäßig verschlüsselte Backups aller Inhalte auf andere Systeme zu erstellen.

2.6.3 Fokus: Identity Provider

Aufgabe des *Identity Provider*-Services ist es, als Basis für vertrauliche Kommunikation sämtliche Informationen rund um eine Persona auf einem selbst-kontrollierten System zu speichern und von dort in verschiedene Nutzungskontexte über Schnittstellen zu integrieren. Ein gesicherter Online-Kontakt kann mit dem *Identity Provider*-Service hergestellt werden, wenn beide Personen einen *Identity Provider*-Service betreiben, der wie auch der SCS-Service eine OAuth-Schnittstellenautorisierung anbietet. Wenn diese sich dann gegenseitig über die OAuth-Schnittstelle Zugriff auf die im *Identity Provider*-Service abgelegten Persona-Informationen autorisieren, d.h. in dieser Situation jeweils als als OAuth-Provider und -Client gleichzeitig auftreten, und ihre Echtheit über die digitale Signatur und die Signatur ihrer Domain gewährleisten, kann ein Kontakt als gesichert angesehen werden. Vertrauliche Kommunikation kann dann unter Zuhilfenahme aller Komponenten der Architektur stattfinden.

2.6.4 Fokus: Vertrauliches Teilen im Web

Zwei Möglichkeiten vertrauliches Teilen stattfinden zu lassen bietet die Architektur. Das Teilen kann in einfacher Form allein mit der Nutzung der bisher beschriebenen Architektur (*Signed Content Storage* + *Identity Provider*) erfolgen oder in womöglich funktionsstärkerem Umfang durch die Anbindung von *Distributed Social Networks* über die *Trusted API*, die den *Signed Content Storage* lediglich als Datenhaltung in ihr eigenes System integrieren.

Um Inhalte vertraulich allein unter Zuhilfenahme des *Signed Content Storages* und des *Identity Providers* zu teilen, bietet dieser Service die Möglichkeit Inhalte nur für autorisierte Kontakte zu veröffentlichen und nicht in öffentlichen Social Networks. Diesen Inhalten können Kontakte aus dem *Identity Provider*-Datenbestand zugeordnet werden, die sich zuvor per OAuth authentifiziert haben. Um Inhalte einsehen zu können, müssen sie sich dann gegenüber des *Signed Content Storages* authentifizieren. Es entsteht so ebenfalls eine Art *Distributed Social Network*, wobei immer Peers miteinander per OAuth abgesichter miteinander kommunizieren, die jeweils einen *Signed Content Storage*- sowie einen *Identity Provider*-Service betreiben.

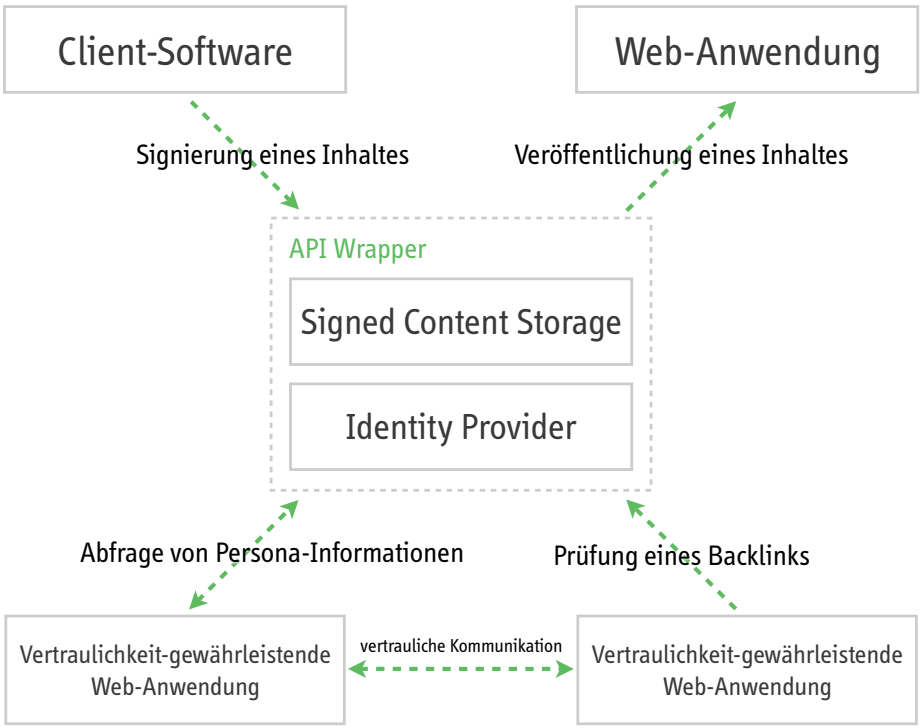

Abbildung 2.18: Übersicht über die beschriebene Architektur im Web-Kontext mit Anwendungsbeispielen

2.7 Evaluation

Ein bekannte konzeptionelle „Lücke“ des Goal-directed Ansatzes ist die nicht vorhandene explizite Benennung einer Evaluationsmethode der darin erschaffenen Entwurfsergebnisse. Durch die konsequente Nutzung von Personas scheint den Autoren offensichtlich ausreichend Reflektion der Nutzerziele vorhanden und eine dedizierte Evaluationsphase überflüssig.

Zwar wurde auch im Rahmen dieser Ausarbeitung reger Gebrauch der Methoden gemacht, die der Goal-directed Ansatz unmittelbar anbietet, um sich die Benutzerziele jederzeit wieder vor Augen zu führen und aktuelle Arbeiten damit abzugleichen, doch wurden in Ergänzung dazu zwischen den Phasen *Modeling* und *Design Framework* sowie zwischen *Design Framework* und *Design Refinement* verschiedene formative Evaluationsmethoden eingeschoben, um auf der Basis dieser Erkenntnisse zunächst überhaupt einen Architekturentwurf skizzieren zu können und in der *Refinement*-Phase diesen noch einmal korrigieren und ausschmücken zu können.

Zwischen *Modeling* und *Design Framework* wurden zunächst verschiedene Architekturentwürfe in Gesprächsgruppen aus Domänenexperten kritisch hinterfragt. Fundamentales Ergebnis dieser Gespräche war die Aufgabe des Ziels ein geschlossenes System aus dem Bereich *Distributed Social Networking* zu entwerfen hin zu der Idee einer Architektur, die web-weite Anwendung finden soll und sich in erster Linie zum Ziel macht, bestehende Services und Problemlösungen zu integrieren.

Zwischen *Design Framework* und *Design Refinement* wurden auf Basis der ersten Architekturentwürfe schließlich im Stile eines *Cognitive Walkthroughs* Personen aus dem Kreis der befragten potentiellen Nutzer mit Prototypen der Architektur konfrontiert. Diese lagen zu diesem Zeitpunkt in gezeichneter Skizzenform und schematischen Abbildungen vor. Sie wurden gebeten verschiedene gestellte Aufgaben unter Zuhilfenahme der Funktionen der Architektur-Services zu erfüllen. Darunter fielen einfachere Aufgaben wie das Signieren einer Kurznachricht, als auch schwierigere Aufgaben, wie die Überprüfung der Integrität eines beim Foto-Diesnt Flickr[55] veröffentlichen Bildes, wobei der Backlink auf das SCS-System als Wasserzeichen im Bild selbst platziert war. Die Ergebnisse dieser Aktivitäten spiegeln sich in sämtlichen Architekturbeschreibungen wider.

Eine abschließende summative Evaluation steht aus, da zu diesem Zwecke der implementierte Prototyp der Referenzimplementierung noch umfangreicher ausgearbeitet werden muss. Die Proof-of-Concept-Stufe, die derzeit ist erreicht ist ermöglicht eine Bewertung der grundlegenden Machbarkeit, jedoch nicht keine gesicherte Aussage

[55] http://www.flickr.com/

über die Vollständigkeit der Anforderungserfüllung, auch wenn die Skizze zeigt, dass sämtliche Benutzeranforderungen ihren Weg in die Architektur gefunden haben.

3 Implementierung

Im Rahmen dieser Ausarbeitung wurde ein vertikaler Prototyp der Referenzimplementierung umgesetzt. Dieser umfasst in einem Beispielkontext im Zusammenspiel mit der Web-Anwendung Twitter die wesentlichen Merkmale des *Signed Content Storages*:

- eine Benutzer-initiierte Autorisierung des *Signed Content Storages* zur Veröffentlichung von Inhalten bei Twitter durch die Implementierung der OAuth-Schnittstelle von Twitter
- eine digitale Signierung neuer Tweets innerhalb des *Signed Content Storages* unter der Nutzung von SHA1-Backlinks
- eine Definition von Nutzungslizenzen pro Inhalt
- eine Veröffentlichung neuer, signierter Tweets über die OAuth-basierte Schnittstelle bei Twitter

Der *Signed Content Storage* agiert somit als OAuth-Client, dessen Rechte in den Einstellungen des Twitter-Accounts eingestellt werden können, der zur Autorisierung genutzt wurde. Die Implementierung erfolgte in der Programmiersprache Ruby[1] (Version 1.8.7) unter Nutzung des Rails-Frameworks[2] (Version 3.0.3) und einer einfachen, relationalen SQLite-Datenbank[3]. Nennenswerte eingesetzte *RubyGems* sind *OmniAuth*[4] als ein Wrapper um die OAuth-Schnittstelle von Twitter und das „Twitter"-Gem als Wrapper um die REST-Schnittstelle zum Veröffentlichen neuer Tweets[5]. Die übrigen verwendeten RubyGems sind im *Gemfile* der Anwendung ersichtlich.

Der Quelltext des Prototypen liegt der Ausarbeitung auf der im Anhang A.4 befindlichen CD-ROM bei. Um die Anwendung auf einem Mac OS X-System zu starten, wird die Nutzung des *Ruby Version Managers „rvm"*[6] in Verbindung mit dem *Dependency Management Tool „Bundler"*[7] empfohlen.

[1] http://ruby-lang.org/
[2] http://rubyonrails.org/
[3] http://www.sqlite.org/
[4] https://github.com/intridea/omniauth
[5] https://github.com/jnunemaker/twitter
[6] http://rvm.beginrescueend.com/
[7] http://gembundler.com/

4 Reflexion und Ausblick

Inhalt und Ziel dieser Ausarbeitung ist die Beantwortung der Frage ob eine Architektur für das Web skizziert werden kann, welche die Schutzziele der IT-Sicherheit Integrität, Verfügbarkeit und Vertraulichkeit im Web geteilter Inhalte gewährleistet und dabei auch durch seine Nutzer als vertrauenswürdig erkannt wird. Hierzu wurde zunächst eine umfangreiche Recherche- und Modellierungsphase durchgeführt, in der sämtliche relevanten Konzepte der Problemdomäne im Detail aufgearbeitet und bewertet wurden. Es wurde eine begriffliche Basis geschaffen auf der Benutzerpartizipation mit zwölf ausgewählten Vertretern potentieller Nutzer der Architektur stattgefunden hat. Die Erkenntnisse sind in greifbare Modelle und schließlich Anforderungen an eine Architektur überführt worden. Die Forschungsfrage lässt sich nun wie folgt beantworten:

Ja, es kann eine Architektur skizziert werden, die auf technischer Ebene Vertraulichkeit, Verfügbarkeit und Integrität von nutzergenerierten Inhalten absichert. Sie kann sogar so skizziert werden, dass diese Schutzziele unabhängig voneinander Anwendung finden können, damit eine zentrale Benutzeranforderung erfüllt wird und die Architektur durch die Möglichkeit einer web-weiten Nutzung an Qualität gewinnt. Die Idee zwei sehr einfache und im Funktionsumfang jeweils fokussiert entwickelte Web-Services über offene Programmierschnittstellen nicht nur mit bestehenden Social Networks zu verbinden, sondern darüber auch die Integration von Anwendungen aus dem Bereich des *Distributed Social Networking* zu ermöglichen, eröffnet eine vollkommen freie Ausgestaltung der Architektur für einen jeden Benutzer. Es werden keine Benutzungsschnittstellen wie Web-Frontends oder Client-Software vorgegeben, sondern die Nutzung zahlreicher verschiedener Software ermöglicht.

Inwiefern die skizzierte Architektur durch die dafür entworfenen Funktionen tatsächlich als vertrauenswürdig aufgefasst wird, dies ist durch eine noch ausstehende summative Evaluation noch zu beweisen. Doch sind die Erkenntnisse aus diesem Bereich, wie sich Vertrauen bildet, wie Vertrauen operationalisiert werden kann und welche Merkmale einer Architektur zur Bildung von Vertrauen beitragen, unmittelbar in die Gestaltung der Architektur eingeflossen. Das Manifest zur Architektur setzt auf Vertrauen als Fundament zur Einhaltung von Regeln und positioniert dieses Verständnis auf diese Weise prominent in der Diskussion um die hier beschriebene Architektur.

Ziel zukünftiger Arbeiten sollte trotzdem sein, Konzepte zur Schaffung von Vertrau-

enswürdigkeit auf der gleichen web-weiten Ebene zu entwickeln, auf der in dieser Ausarbeitung die Gewährleistung der Schutzziele diskutiert wird.

Ferner stehen auf technischer Seite Weiter- und Neuentwicklungen der Referenzimplementierungen an. Die in dieser Ausarbeitung dargestellte Implementierung ist lediglich ein vertikaler Proof-of-Concept-Prototyp und berücksichtigt noch nicht alle Aspekte der skizzierten Architektur. Darüber hinaus ist sie eine Ruby-Implementierung, sollte jedoch im Zuge der breiten und einfachen Nutzbarkeit und Installation auch in weiteren populären Programmiersprachen für das Web (z.B. PHP oder Python) umgesetzt werden. Auch ist im Zuge der Weiterentwicklung eine Integration weiterer offener Web-Standards interessant, um die universelle Nutzung und die stärkere Integration in den Web-Kontext zu verbessern. Hier erscheint vor allem eine Implementierung der *OStatus*-Spezifikation[1] für das Echtzeit-Routing von signierten Kurznachrichten im Web interessant, aber auch das *webfinger*-Projekt[2] könnte im Rahmen des *Signed Content Storages* Verwendung finden, um darüber auf den selbst-betriebenen *Storage Service* aufmerksam zu machen oder um die Echtheit des Services anhand eines weiteren Kriteriums zu belegen.

Für die Nutzung der angebotenen *Trusted API*, also der Schnittstelle beider Services der Architektur, sollten Bibliotheken und *Wrapper* für verschiedene populäre Programmiersprachen entwickelt werden (Ruby, PHP, iOS, Android, Java), um auch die Entwicklung mobiler Clients voranzutreiben.

Abschließend soll ein Blick auf die beiden in der Einführung formulierten Hypothesen gerichtet werden.

Hypothese 1 *Eine Web-Anwendung kann nur dann die Schutzziele der IT-Sicherheit vollständig gewährleisten, wenn sie nicht zentralistisch organisiert ist, sondern die Kontrolle über die Speicherung und Vermittlung der eigenen Daten allein bei deren Urhebern liegt.*

Dieser These ist auf Basis der Erkenntnisse dieser Ausarbeitung zuzustimmen. Auch die hier beschriebene Architektur stellt aus höher betrachteter Perspektive über die OAuth-Schnittstellen ein dezentrales Social Network dar, auch wenn sie dabei eine andere (web-weite) Ausrichtung besitzt.

Hypothese 2 *Die Bereitschaft eine Web-Anwendung zu nutzen, die auf einer alternativen Architektur fußt, geht nur dann in eine tatsächliche Nutzung über, wenn die Anwendung*

[1] http://ostatus.org/
[2] http://code.google.com/p/webfinger/

*selbst eine **nachvollziehbare Gewährleistung der Schutzziele** seitens der Nutzer ermöglicht. Eine Architektur zum Teilen nutzergenerierter Inhalte im Web sollte aus sich heraus **vertrauenswürdig** sein, um Motivation für eine Nutzung zu schaffen.*

Diese Annahme ist im Laufe der Ausarbeitung zu einer gefestigten Erkenntnis gereift: Eine Architektur hat diese Eigenschaft aufzuweisen und es sind Möglichkeiten dargestellt sie auszubilden. Inwiefern diese jedoch tatsächlich eine spätere Nutzung beeinflussen, das wird Ergebnis einer sich noch anschließenden Evaluation sein.

Anhang

A.1 Wertewelten von Digital Visitors und Digital Residents nach Kruse

Abbildung A.1: Wertewelt der „Digital Visitors“

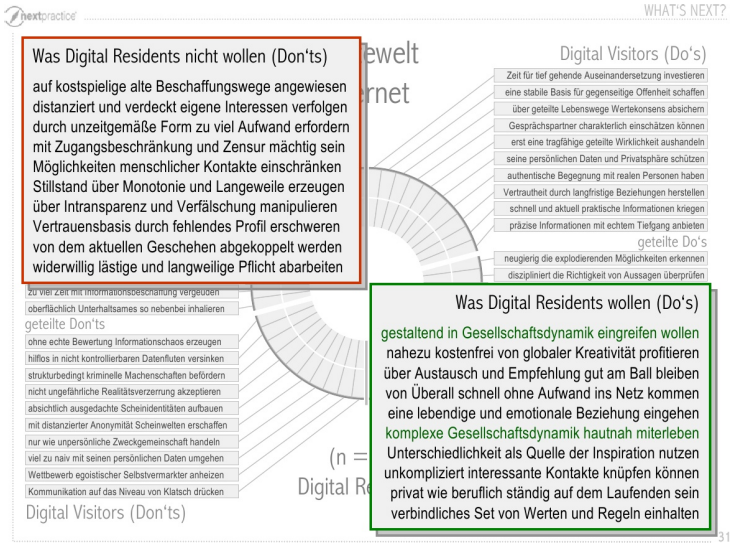

Abbildung A.2: Wertewelt der „Digital Residents“

A.2 Steckbriefe befragter Personen im Rahmen der Benutzerpartizipation

Kennung	**EL**
Beruf	Designer
Geschlecht	männlich
Alter	30
Beschreibung	kennt das aktuelle Web sehr gut, benutzt viele Social Services, um einfach und vorwiegend mobil per iPhone Informationen (überwiegend Geo, Status, Fotos) zu veröffentlichen. Produziert viele Inhalte, hält allerdings die Reaktionen darauf nicht nach und konsumiert weniger. Extraviert.
Nutzungsrolle: Produzent vs. Konsument	eher Produzent
Nutzungsmotivation: privat vs. beruflich	privat
Nutzungskontext: stationär vs. mobil	mobil
Aktuelle Nutzung: ja vs. nein	ja
Domänenwissen: vorhanden vs. nicht vorhanden	nicht vorhanden
Inhalte	Status, Geo, Fotos, Links
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Öffentlichkeit
Hinreichende Sicherheit	geringe Sicherheit genügt
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	Digital Resident

Abbildung A.3: Steckbrief der befragten Person EL

Kennung	**MN**
Beruf	Web-Entwickler
Geschlecht	männlich
Alter	28
Beschreibung	kennt das aktuelle Web sehr gut, hat auch beruflich damit zu tun. Nutzt es gleichermaßen professionell wie privat. Veröffentlicht gerne Fotos von seinem iPhone und teilt hin und wieder Kurznachrichten mit seinen Freunden. Kennt die „Gefahren“ des öffentlichen Teilens im Web, ist jedoch bereit sie einzugehen, um am öffentlichen Leben im Web teilzunehmen
Nutzungsrolle: Produzent vs. Konsument	Produzent/Konsument
Nutzungsmotivation: privat vs. beruflich	beruflich
Nutzungskontext: stationär vs. mobil	mobil
Aktuelle Nutzung: ja vs. nein	ja
Domänenwissen: vorhanden vs. nicht vorhanden	vorhanden
Inhalte	Status, Geo, Fotos
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Öffentlichkeit
Hinreichende Sicherheit	geringe Sicherheit genügt
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	Digital Resident

Abbildung A.4: Steckbrief der befragten Person MN

Kennung	**KE**
Beruf	Web-Entwickler
Geschlecht	männlich
Alter	28
Beschreibung	steht dem öffentlichen Teilen im Web kritisch gegenüber und agiert stets unter Pseudonym, um keine Auswirkungen auf seine Reputation befürchten zu müssen. Nichtsdestotrotz teilt er überdurchschnittlich oft Status und Links in der Öffentlichkeit unter Pseudonym und in der Teilöffentlichkeit unter seiner wahren Identität
Nutzungsrolle: Produzent vs. Konsument	Produzent/Konsument
Nutzungsmotivation: privat vs. beruflich	privat
Nutzungskontext: stationär vs. mobil	stationär
Aktuelle Nutzung: ja vs. nein	ja
Domänenwissen: vorhanden vs. nicht vorhanden	vorhanden
Inhalte	Links, Videos
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Teilöffentlichkeit
Hinreichende Sicherheit	hohe Sicherheit notwendig
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	Digital Visitor

Abbildung A.5: Steckbrief der befragten Person KE

Kennung	**KL**
Beruf	Web-Entwickler
Geschlecht	männlich
Alter	28
Beschreibung	besitzt einen niedrigen Anspruch an Datensicherheit im Netz, auch wenn er dies prinzipiell begrüßen würde. Hat aber auch kein Problem damit in unbekannten Kontexten unter seiner wahren Identität aufzutreten und nutzt regelmäßig, wenn auch nicht oft, viele Social Services. Beruflich nutzt er die Möglichkeiten des Webs, um Inhalte mit Kunden zu teilen und gemeinsam mit seinem Team zu diskutieren
Nutzungsrolle: Produzent vs. Konsument	eher Konsument
Nutzungsmotivation: privat vs. beruflich	beruflich
Nutzungskontext: stationär vs. mobil	stationär
Aktuelle Nutzung: ja vs. nein	ja
Domänenwissen: vorhanden vs. nicht vorhanden	vorhanden
Inhalte	Status
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Öffentlichkeit
Hinreichende Sicherheit	geringe Sicherheit genügt
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	Digital Resident

Abbildung A.6: Steckbrief der befragten Person KL

Kennung	**DJ**
Beruf	Kommunikationsberater
Geschlecht	männlich
Alter	43
Beschreibung	nutzt das Web vorwiegend beruflich. Teilt in erster Linie berufliche Dokumente, die stets eine gewisse Vertraulichkeit aufweisen und deren Veröffentlichung seinem Geschäft Schaden zufügen könnte. Primäres Werkzeug dazu ist E-Mail, obwohl er weiß, dass E-Mail kein sicheres Kommunikationswerkzeug ist. Er hat Interesse an Alternativen, vor allem wenn diese eine mobile Nutzung geteilter Inhalte ermöglichen
Nutzungsrolle: Produzent vs. Konsument	Produzent/Konsument
Nutzungsmotivation: privat vs. beruflich	beruflich
Nutzungskontext: stationär vs. mobil	mobil
Aktuelle Nutzung: ja vs. nein	nein
Domänenwissen: vorhanden vs. nicht vorhanden	vorhanden
Inhalte	Dokumente
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Teilöffentlichkeit
Hinreichende Sicherheit	hohe Sicherheit notwendig
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	Digital Visitor

Abbildung A.7: Steckbrief der befragten Person DJ

Kennung	**TD**
Beruf	Theologe
Geschlecht	männlich
Alter	65
Beschreibung	besitzt wenig technische Expertise und nutzt das Web zögerlich, entdeckt aber nach und nach die Möglichkeiten mit Freude. Die Frage hinsichtlich der Vertraulichkeit eigener Inhalte war neu für ihn, dann jedoch ein gern diskutierter Aspekt. Möchte gerne dazulernen und Wege kennenlernen, die ihm vertrauliches Teilen bei einfacher Bedienung ermöglichen
Nutzungsrolle: Produzent vs. Konsument	eher Konsument
Nutzungsmotivation: privat vs. beruflich	privat
Nutzungskontext: stationär vs. mobil	stationär
Aktuelle Nutzung: ja vs. nein	nein
Domänenwissen: vorhanden vs. nicht vorhanden	nicht vorhanden
Inhalte	Fotos, Videos, Texte
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Teilöffentlichkeit
Hinreichende Sicherheit	gewisse Sicherheit genügt
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	kein Heavy User

Abbildung A.8: Steckbrief der befragten Person TD

Kennung	**SÖ**
Beruf	Psychologin
Geschlecht	weiblich
Alter	27
Beschreibung	steht dem Agieren im Web unter Klarnamen aus beruflichen Gründen kritisch gegenüber, da Veröffentlichungen zu ihrer Person ihre Arbeit beeinträchtigen können. Handelt daher stets bewusst in sicher abgegrenzter Teilöffentlichkeit, teilt lieber einen Inhalt weniger als zuviel und achtet auch dort auf die genauen Inhalte.
Nutzungsrolle: Produzent vs. Konsument	eher Konsument
Nutzungsmotivation: privat vs. beruflich	privat
Nutzungskontext: stationär vs. mobil	stationär
Aktuelle Nutzung: ja vs. nein	nein
Domänenwissen: vorhanden vs. nicht vorhanden	nicht vorhanden
Inhalte	Fotos, Videos, Texte, Links
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Teilöffentlichkeit
Hinreichende Sicherheit	gewisse Sicherheit genügt
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	Digital Visitor

Abbildung A.9: Steckbrief der befragten Person SÖ

Kennung	**ZK**
Beruf	Betriebswirtin
Geschlecht	weiblich
Alter	35
Beschreibung	nutzt das Web überwiegend privat zu Recherchezwecken und zum Einkaufen. Das Teilen im Web gehört aktuell noch nicht zu ihren primären Tätigkeiten, da sie sich unsicher ist, welche Konsequenzen dies bedeuten würde. Mehr Kontrolle über die eigenen Inhalte würde helfen diese Unsicherheit abzubauen
Nutzungsrolle: Produzent vs. Konsument	eher Konsument
Nutzungsmotivation: privat vs. beruflich	privat
Nutzungskontext: stationär vs. mobil	stationär
Aktuelle Nutzung: ja vs. nein	ja
Domänenwissen: vorhanden vs. nicht vorhanden	nicht vorhanden
Inhalte	Fotos, Links
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Teilöffentlichkeit
Hinreichende Sicherheit	gewisse Sicherheit genügt
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	Digital Visitor

Abbildung A.10: Steckbrief der befragten Person ZK

Kennung	**KN**
Beruf	Web-Entwickler
Geschlecht	männlich
Alter	27
Beschreibung	zwar nutzt er das Web beruflich im professionellen Kontext und besitzt umfangreiches Fachwissen, doch steht der dem Teilen im Web aktuell kritisch gegenüber, agiert stets in Teilöffentlichkeiten unter Pseudonym und versucht eigene Inhalte auf ein Minimum zu reduzieren. Hin und wieder nutzt er das Web mobil, um die geteilten Inhalte seiner Freunde einzusehen und Fotos hochzuladen
Nutzungsrolle: Produzent vs. Konsument	Produzent/Konsument
Nutzungsmotivation: privat vs. beruflich	beruflich
Nutzungskontext: stationär vs. mobil	mobil
Aktuelle Nutzung: ja vs. nein	ja
Domänenwissen: vorhanden vs. nicht vorhanden	vorhanden
Inhalte	Status, Geo, Fotos
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Teilöffentlichkeit
Hinreichende Sicherheit	gewisse Sicherheit genügt
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	Digital Resident

Abbildung A.11: Steckbrief der befragten Person KN

Kennung	**JD**
Beruf	Lehrer/Literat
Geschlecht	männlich
Alter	23
Beschreibung	besitzt durch längere Auslandsaufenthalte Bekannte in ganz Europa und nutzt das Social Web und mit ihnen in Verbindung zu bleiben. Er agiert dabei unter Pseudonym und je nach Kontext unter verschiedenen Personae, teilt jedoch in hoher Frequenz Texte, Fotos und Links und nutzt das Web lieber als Kommunikationmittel unter Freunden als das Telefon oder SMS-Nachrichten
Nutzungsrolle: Produzent vs. Konsument	Produzent/Konsument
Nutzungsmotivation: privat vs. beruflich	privat
Nutzungskontext: stationär vs. mobil	stationär
Aktuelle Nutzung: ja vs. nein	ja
Domänenwissen: vorhanden vs. nicht vorhanden	nicht vorhanden
Inhalte	Geo, Events, Status, Links, Texte
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Teilöffentlichkeit
Hinreichende Sicherheit	geringe Sicherheit genügt
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	Digital Resident

Abbildung A.12: Steckbrief der befragten Person JD

Kennung	**KD**
Beruf	Musiker
Geschlecht	männlich
Alter	32
Beschreibung	ist als Musiker interessiert an fachlichem Austausch über spezifische Themen und teilt Musik oder Arrangements mit Kollegen und Fotos und Videos seiner Auftritte mit der Öffentlichkeit. Obwohl er es als Musiker gewohnt ist in der Öffentlichkeit aufzutreten, begleitet ihn beim Teilen von Inhalten im Web ein Unwohlsein bei dem Gedanken, dass Informationen zu seiner Person umfangreich im Web vorhanden und leicht zu einem Gesamtbild zusammenzufügen sind. Aus diesem Grund hält er das private Leben komplett aus dem Web heraus
Nutzungsrolle: Produzent vs. Konsument	Produzent/Konsument
Nutzungsmotivation: privat vs. beruflich	beruflich, gern aber auch privat
Nutzungskontext: stationär vs. mobil	stationär
Aktuelle Nutzung: ja vs. nein	ja
Domänenwissen: vorhanden vs. nicht vorhanden	nicht vorhanden
Inhalte	Musik, Links, Fotos, Videos
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Teilöffentlichkeit
Hinreichende Sicherheit	gewisse Sicherheit genügt
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	kein Heavy User

Abbildung A.13: Steckbrief der befragten Person KD

Kennung	**KT**
Beruf	Sozialpädagoge
Geschlecht	männlich
Alter	30
Beschreibung	nutzt das Web und darin vorwiegend geschlossene Social Networks ausschließlich privat, um unter Pseudonym mit seinen Freunden in anderen Städten in Kontakt zu bleiben. Die Funktionen, die Social Networks bieten, gefallen ihm jedoch sehr gut. Auch in Teilöffentlichkeit begleitet ihn trotzdem ein Unbehagen durch techn. Unwissenheit was genau mit seinen Inhalten im Web geschieht und ob er jederzeit die Kontrolle darüber hätte. Die Dienste großer Konzerne versucht er zu meiden, um Dateninseln zu reduzieren
Nutzungsrolle: Produzent vs. Konsument	Produzent/Konsument
Nutzungsmotivation: privat vs. beruflich	privat
Nutzungskontext: stationär vs. mobil	stationär
Aktuelle Nutzung: ja vs. nein	ja
Domänenwissen: vorhanden vs. nicht vorhanden	nicht vorhanden
Inhalte	Links, Fotos, Dokumente
Bevorzugte Öffentlichkeit: öffentlich vs. teilöffentlich	Teilöffentlichkeit
Hinreichende Sicherheit	hohe Sicherheit erwünscht
ggfs. Wertepräferenz: Digital Visitors vs. Digital Residents	kein Heavy User

Abbildung A.14: Steckbrief der befragten Person KT

A.3 Schnittstellenbeschreibungen

Die Schnittstelle des *Signed Content Storage*-Services genügt dem REST-Architekturstil. Sie ermöglicht im Wesentlichen den Zugriff auf die Ressource *Content*. Auf ihr ist neben der klassischen *Index*- und *Show*-Methoden vor allem die POST-Methode zum Anlegen neuer Inhalte ausführbar:

	Methode: POST (einen neuen Inhalt anlegen)
Parameter	Beschreibung
title	textueller Titel des Content-Objektes, kann nur intern, aber auch öffentlich genutzt werden
payload	textueller Inhalt des Content-Objektes
attachment	Binärdatum eines Content-Objektes
license	textuelle Definition einer Nutzungslizenz
target	Foreign-Key eines Social Networks, für den ein Inhalt primär bestimmt ist
people	Array aus E-Mail-Adresen oder sonstigen eindeutigen Kennungen der autorisierten Teilhaber (sofern notwendig)

Tabelle A.1: Eingangsparameter der POST-Methode der *Content*-Ressource der Schnittstelle des *Signed Content Storage*-Services

Diese wird primär von Client-Software implementiert und orientiert sich an den aufgestellten datenbezogenen Anforderungen.

Auf der anderen Seite sollten integrierte Web-Anwendungen eine REST-Schnittstelle zur Verfügung stellen, über die vom SCS aus Zugriff auf die entfernte Content-Ressource *RemoteContent* stattfinden kann. Hier ist vor allem die PUT-Methode mit ihrer Bedeutung „Veränderung der Ressource“ und die DELETE-Methode mit ihrer Bedeutung „Löschung der Ressource“ von Interesse.

Bei der PUT-Methode werden zusätzlich zum *identifier* im Wesentlichen erneut alle Attribute eines *Content*-Objektes übertragen.

	Methode: PUT (verändert einen gespeicherten Inhalt)
Parameter	Beschreibung
identifier	eindeutige ID des gespeicherten Inhaltes, der vom SCS bei der Speicherung vorgegeben wird und SHA1-Format aufweist
title	textueller Inhalt des Content-Objektes
payload	textueller Inhalt des Content-Objektes
attachment	Binärdatum eines Content-Objektes
license	textuelle Definition einer Nutzungslizenz
people	Array aus E-Mail-Adresen oder sonstigen eindeutigen Kennungen der autorisierten Teilhaber (sofern möglich)

Tabelle A.2: Eingangsparameter der PUT-Methode der *RemoteContent*-Ressource der Schnittstelle eines in die Architektur integrierten Web-Services

	Methode: DELETE (löscht einen Inhalt)
Parameter	Beschreibung
identifier	eindeutige ID des gespeicherten Inhaltes, der vom SCS bei der Speicherung vorgegeben wird und SHA1-Format aufweist
hard_delete	Boolean; sollte ein integrierter Web-Service verschiedene Arten von Lösung kennen (z.B. einen *Soft Delete*, bei dem ein Inhalt lediglich als gelöscht markiert wird und einen *Hard Delete*, bei den dies tatsächlich auch in der Datenbank passiert), dann kann über diesen Parameter ein Hard Delete erzwungen werden.

Tabelle A.3: Eingangsparameter der DELETE-Methode der *RemoteContent*-Ressource der Schnittstelle eines in die Architektur integrierten Web-Services

A.4 CD-ROM

Auf der beiliegenden CD-ROM befindet sich der Quellcode der in Abschnitt 3 beschriebenen Referenzimplementierung. Systemvoraussetzung ist eine Ruby 1.8.7-Umgebung mit den im Gemfile der Anwendung angegebenen installierten RubyGems.

Verzeichnisse

Literaturverzeichnis

[BH98] BEYER, HUGH KAREN HOLTZBLATT: *Contextual Design: Defining Customer-Centered Systems (Interactive Technologies)*. Morgan Kaufmann, 1st , 9 1998.

[CRC07] COOPER, ALAN, ROBERT REIMANN ROBERT CRONIN: *About Face 3: The Essentials of Interaction Design*. Wiley, 3rd , 5 2007.

[DFAB03] DIX, ALAN, JANET E. FINLAY, GREGORY D. ABOWD RUSSELL BEALE: *Human-Computer Interaction (3rd Edition)*. Prentice Hall, 3 , 12 2003.

[fECoD07] ECONOMIC CO-OPERATION, ORGANISATION FOR DEVELOPMENT: *Participative Web and User-Created Content. Technical report.* 2007.

[fS05] STANDARDIZATION, INTERNATIONAL ORGANIZATION FOR: *ISO 9000 Quality management systems – Fundamentals and vocabulary*. 2005.

[HV09] HAGEMANN, STEPHAN GOTTFRIED VOSSEN: *Categorizing User-Generated Content*. 2009.

[HW01] HOUSER, DANIEL JOHN WOODERS: *Reputation in Auctions: Theory and Evidence from eBay*. Mimeo, 2001.

[Lam95] LAMNEK, SIEGFRIED: *Qualitative Sozialforschung, 2 Bde., Bd.1, Methodologie*. BeltzPVU, 12 1995.

[Luh00] LUHMANN, NIKLAS: *Vertrauen. Ein Mechanismus der Reduktion sozialer Komplexität.* UTB, Stuttgart, 12 2000.

[May99] MAYHEW, DEBORAH J.: *The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design (Interactive Technologies)*. Morgan Kaufmann, 1st , 4 1999.

[Pet96] PETERMANN, FRANZ: *Psychologie des Vertrauens*. Hogrefe, 3., korrigierte Aufl , 1996.

[RC01] ROSSON, MARY BETH JOHN M. CARROLL: *Usability Engineering: Scenario-Based Development of Human-Computer Interaction (Interactive Technologies)*. Morgan Kaufmann, 1st , 10 2001.

[RZSL06] RESNICK, PAUL, RICHARD ZECKHAUSER, JOHN SWANSON KATE LOCKWOOD: *The value of reputation on eBay: A controlled experiment*. Experimental Economics, 9(2):79–101, 2006.

[SB92] SCHAUMÜLLER-BICHL, INGRID: *Sicherheitsmanagement*. BI-Wissenschaftsverlag, Mannheim, 1992.

Internetquellen

[Bun86] BUNDESRAT, DEUTSCHER: *Geheimschutzordnung des Bundesrates*. http://www.bundesrat.de/cln_171/nn_9720/SharedDocs/Gesetzestexte/geheimschutzordnung__bundesrat,templateId=raw,property=publicationFile.pdf/geheimschutzordnung_bundesrat.pdf, November 1986. zuletzt abgerufen am 30.12.2010.

[Bun09] BUNDESVERFASSUNGSGERICHT: *Verwendung von Wahlcomputern bei der Bundestagswahl 2005 verfassungswidrig*. http://www.bundesverfassungsgericht.de/pressemitteilungen/bvg09-019, 2009. zuletzt abgerufen am 30.12.2010.

[Bun10] BUNDESTAG, DEUTSCHER: *Grundgesetz der Bundesrepublik Deutschland, I. Die Grundrechte*. http://www.bundestag.de/dokumente/rechtsgrundlagen/grundgesetz/gg_01.html, Juli 2010. zuletzt abgerufen am 30.12.2010.

[Cru10] CRUNCHBASE: *StudiVZ*. http://www.crunchbase.com/product/studivz, November 2010. zuletzt abgerufen am 30.12.2010.

[Das10] DASERSTE.DE: *Wirbel um Wikileaks-Enthüllung - peinliches Zeugnis für Schwarz-Gelb?* http://daserste.ndr.de/annewill/archiv/erste10593.html, November 2010. zuletzt abgerufen am 30.12.2010.

[Dro10] DROPBOX: *Dropbox Security Overview*. http://www.dropbox.com/terms#security, 2010. zuletzt abgerufen am 30.12.2010.

[Eur10] EUROPE, HOST: *Service Level Agreements - Servicegarantien für Host Europe Produkte*. http://www.hosteurope.de/content/Webhosting-SLA, November 2010. zuletzt abgerufen am 30.12.2010.

[Fac10] FACEBOOK: *Facebook Statistiken*. http://www.facebook.com/press/info.php?statistics, November 2010. zuletzt abgerufen am 30.12.2010.

[Hae06] HAEUSLER, JOHNNY: *Grimme für Spreeblick!* http://www.spreeblick.com/2006/06/02/grimme-fur-spreeblick/, Juni 2006. zuletzt abgerufen am 30.12.2010.

[Hed10] HEDEMANN, FALK: *Diaspora: Welche Chancen hat die dezentrale Facebook-Alternative?* t3n Magazin, http://t3n.de/news/diaspora-welche-chancen-hat-dezentrale-279997/, 2010. zuletzt abgerufen am 30.12.2010.

[Hei10] HEINE, FRANZISKA: *Petition: Internet - Keine Indizierung und Sperrung von Internetseiten vom 22.04.2009*. https://epetitionen.bundestag.de/index.php?action=petition;sa=details;petition=3860, Juni 2010. zuletzt abgerufen am 30.12.2010.

[Her09] HERZOG, MARTINA: *Romantik 2.0 - Ein Heiratsantrag auf Twitter*. Der-Westen, http://www.derwesten.de/nachrichten/technik/Romantik-2-0-Ein-Heiratsantrag-auf-Twitter-id277999.html, April 2009. zuletzt abgerufen am 30.12.2010.

[ho02] ONLINE HEISE: *Wie man SIM-Karten fälscht*. http://www.heise.de/newsticker/meldung/Wie-man-SIM-Karten-faelscht-58333.html, 2002. zuletzt abgerufen am 30.12.2010.

[ho06] ONLINE HEISE: *Datenleck beim StudiVZ? [Update]*. http://www.heise.de/newsticker/meldung/Datenleck-beim-StudiVZ-Update-119903.html, 2006. zuletzt abgerufen am 30.12.2010.

[Jou84] JOURNAL, ZDF HEUTE: *BTX-Hack*. http://video.google.com/googleplayer.swf?docId=-8396178892678063881&hl=de, 1984. zuletzt abgerufen am 30.12.2010.

[Kru10] KRUSE, PETER: *What's next? Die die Netzwerke Wirtschaft und Gesellschaft revolutionieren*. http://www.scribd.com/doc/29900810/republica2010, April 2010. zuletzt abgerufen am 30.12.2010.

[Mö10] MÖLLERING, GUIDO: *Das Aufheben von Ungewissheit als Kern des Vertrauens: Just do it?* Max-Planck-Institut für Gesellschaftsforschung, http://www.mpifg.de/pu/workpap/wp06-5/wp06-5.html, November 2010. zuletzt abgerufen am 30.12.2010.

[Net10] NETWORKS, MTV: *Crewsers, Funatics und Mediacs: Volkswagen und MTV Networks veröffentlichen internationale Social Media Studie "MePublic"*. http://pressecenter.mtv.de/channel/1,2,3,4,5/83770, September 2010. zuletzt abgerufen am 30.12.2010.

[Pri10] PRITLOVE, TIM: *CRE 30K*. http://blog.chaosradio.ccc.de/index.php/2010/04/06/cre-30k/, April 2010. zuletzt abgerufen am 30.12.2010.

[Ras10a] RASKIN, AZA: *The 7 Things That Matter Most in Privacy*. http://www.azarask.in/blog/post/what-should-matter-in-privacy/, 2010. zuletzt abgerufen am 30.12.2010.

[Ras10b] RASKIN, AZA: *Making Privacy Policies not Suck*. http://www.azarask.in/blog/post/making-privacy-policies-not-suck/, 2010. zuletzt abgerufen am 30.12.2010.

[RWC10] REZEPTE-WIKI-COMMUNITY: *Rezepte-Wiki*. http://www.rezeptewiki.org/wiki/Hauptseite, November 2010. zuletzt abgerufen am 30.12.2010.

[Sch10a] SCHMITZ, GREGOR PETER: *Datendesaster erschüttert Washington*. http://www.spiegel.de/politik/ausland/0,1518,731694,00.html, November 2010. zuletzt abgerufen am 30.12.2010.

[Sch10b] SCHONFELD, ERICK: *StatusNet (Of Identi.ca Fame) Raises 875,000 USD To Become The WordPress Of Microblogging*. TechCrunch, http://t.co/jeY5QLY, 2010. zuletzt abgerufen am 30.12.2010.

[Sie10a] SIEGLER, MG: *Diaspora's Final Tally: 200,000 USD From Nearly 6,500 Backers*. TechCrunch, http://tcrn.ch/dfP6Zm, 2010. zuletzt abgerufen am 30.12.2010.

[Sie10b] SIEGLER, MG: *Facebook Competitor Diaspora Revealed: Sparse, But Clean; Source Code Released*. TechCrunch, http://techcrunch.com/2010/09/15/diaspora-revealed/, 2010. zuletzt abgerufen am 30.12.2010.

[ste07] STERN.DE: *20 Jahre "KGB-Hack": Wie 75 Cent zum Verhängnis wurden*. http://t.co/0kQqKMq, 2007. zuletzt abgerufen am 30.12.2010.

[Stu10] STUDIVZ: *Verlass dich drauf – die VZ-Netzwerke sind TÜV SÜD geprüft*. http://www.studivz.net/l/zertifikat, 2010. zuletzt abgerufen am 30.12.2010.

[W3C04] W3C: *Web Services Architecture - W3C Working Group Note 11 February 2004*. http://www.w3.org/TR/ws-arch/#id2260892, 2004. zuletzt abgerufen am 30.12.2010.

[Wal10] WALES, JIMMY: *Ein persönlicher Aufruf von Wikipedia-Grüner Jimmy Wales*. https://spenden.wikimedia.de/spenden/, November 2010. zuletzt abgerufen am 30.12.2010.

[Wau10] WAUTERS, ROBIN: *Dropbox Announces 4 Million Users, Hires A VP From Salesforce*. TechCrunch.com, http://techcrunch.com/2010/01/20/dropbox-4-million-user/, 2010. zuletzt abgerufen am 30.12.2010.

[Wei10] WEIGERT, MARTIN: *Alternative Social Networks wittern ihre Chance*. NETZWERTIG.COM, http://t.co/nMiwsTa, 2010. zuletzt abgerufen am 30.12.2010.

[Wik10] WIKILEAKS: *Secret US Embassy Cables*. http://wikileaks.info/, November 2010. zuletzt abgerufen am 30.12.2010.

[WO08] WELT-ONLINE: *Youtube und Facebook sind die neuen Giganten*. http://www.welt.de/wirtschaft/webwelt/article2904246/Youtube-und-Facebook-sind-die-neuen-Giganten.html, Dezember 2008. zuletzt abgerufen am 30.12.2010.

Tabellenverzeichnis

Abbildungsverzeichnis

Erklärung über die selbständige Abfassung der Arbeit

Ich versichere, die von mir vorgelegte Arbeit selbständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Köln, 30. Dezember 2011

Tim Schneider